THE USE OF MICROWAVE DEVICES IN ELECTRONIC DIGITAL COMPUTERS

ROY D. SNYDER, JR.

U. S. NAVAL POSTGRADUATE SCHOOL
MONTEREY, CALIFORNIA

IN REPLY REFER TO:

NC4(14)
30 June 1958

MEMORANDUM

From: OinC, Engrg Electronics
To: Librarian

Subj: Proprietary reservation on thesis by LT Roy D. SNYDER, USN, from General Electric Company

1. Mr. H. M. Dustin of the Phoenix office of the General Electric Company Industrial Computer Department, has this date informed me by telephone that the thesis by LT Roy D. SNYDER contains at least ten points relating to items of proprietary interest to the General Electric Company and upon which patent proceedings are pending. Therefore, in accordance with the terms of the agreement made with the GE Company by the Navy, he has requested that distribution of this thesis be withheld and that all copies in our possession be placed in a secure locked storage of restricted access.

2. In accordance with the personal agreement between LT Snyder and the GE Company, the outside cover of the thesis was to have prominently and permanently affixed thereto the following restrictive legend:

"FOR OFFICIAL USE ONLY"
"This document contains confidential and proprietary information of the General Electric Company. Any person who accepts this document agrees to refrain from further transferring this document and from disclosing to others any of the proprietary information of the General Electric Company contained therein."

If this legend does not appear on the outside cover, will you please have it placed on all copies in your possession. I will notify LT Snyder and Professor Cotton to do likewise.

L. P. SPEAR

(hh)

THE USE OF MICROWAVE DEVICES IN ELECTRONIC DIGITAL COMPUTERS

* * * * * * * * * *

THE USE OF MICROWAVE DEVICES IN ELECTRONIC DIGITAL COMPUTERS

by

Roy D. Snyder, Jr.
//
Lieutenant, United States Navy

Submitted in partial fulfillment of
the requirements for the degree of

MASTER OF SCIENCE
IN
ENGINEERING ELECTRONICS

United States Naval Postgraduate School
Monterey, California
1958

THE USE OF MICROWAVE DEVICES IN ELECTRONIC DIGITAL COMPUTERS

by

Roy D. Snyder, Jr.

This work is accepted as fulfilling
the thesis requirements for the degree of

MASTER OF SCIENCE

IN

ENGINEERING ELECTRONICS

from the
United States Naval Postgraduate School

ABSTRACT

This paper discusses techniques for utilizing the high operating frequencies and broad pass band of microwave components to store and arithmetically manipulate numbers in digital computers. Gating elements to respond to the presence or absence of an r.f. pulse are mentioned fleetingly. Proposed storage and gating devices which respond to a phase script wherein a pulse of certain r.f. phase represents a "one" and a pulse of opposite r.f. phase represents a "zero" are described. Logic circuits using these devices are proposed for an arithmetic unit inherently capable of algebraic addition of two 20 bit numbers in less than 100 musec and multiplying such numbers in about 1.5 musec. Previous work done with Frequency Domain Techniques is reviewed. Circuits are described whereby arithmetic operations on numbers are performed by operation on groups of radio-frequency pulses with selected frequencies that represent these numbers to form other groups of pulses with frequencies that represent the sums differences, etc., of the input numbers. Traveling wave tube characteristics are investigated relative to the feasibility of designing a short delay (of the order of 1 musec), high gain (greater than 20 db) tube.

The writer wishes to thank Professor M. L. Cotton of the U.S. Naval Postgraduate School for his encouragement and professional counsel in the preparation of this paper while acting as faculty adviser, and Dr. M. W. Bauer for his valuable assistance as second reader. Particular thanks are also due to Dr. R. R. Johnson, Manager of the General Electric Industrial Computer Laboratory, Palo Alto, California, where much of the investigation for this paper was carried out, for his encouragement and helpful suggestions, and to M. P. Forrer of the General Electric Microwave Laboratory for his invaluable assistance.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Cont'd)

TABLE OF SYMBOLS AND ABBREVIATIONS

mcs	megacycles per second
mμsec	millimicrosecond
r-f	radio frequency
kmc	kilomegacycles per second
μsec	microsecond
d.c.	direct current
db	decibel
ω	angular frequency in radians per second
e/m	electron charge to mass ratio
x	magnetic flux
λ	wavelength
f	frequency
TWT	traveling-wave tube
TWA	traveling-wave amplifier
CW	continuous wave
p.r.f.	pulse repetition frequency
BPF	band pass filter
MFM	multifrequency memory
HPF	high pass filter
LPF	low pass filter
G	traveling-wave tube gain
C	a gain parameter
t_d	delay time
K	helix impedance factor
I_o	beam current

V_o	beam voltage
β_o	free space phase constant
β	axial phase constant of helix wave
γ	radial propagation constant
v	phase velocity
c	velocity of light 3×10^8 m/sec
a	mean helix radius
ψ	helix pitch angle
$\cot \psi$	cotangent of helix pitch angle

CHAPTER I

INTRODUCTION

The tremendous growth of electronic computers in size, complexity and capability in recent years has been phenomenal. The enormous development of electronic technology during World War II provided the impetus for this growth which saw numerous computer projects start around a nucleus of wartime radar experts. Electronics not only provided the technological means for greatly increased speed and capacity, and thereby enhanced the usefulness of computers many times, but the availability of cheap, mass-produced components and of engineers trained to use them made it possible to experiment on a greater scale and at a lower capital investment than before. This led to the development of new components and methods of construction and computer engineering was firmly implanted as a major division of the electronics industry.

As electronic computers have grown, operating speeds have steadily increased so that the concept of high speed has become a purely relative one. When the first electronic computers emerged in the late 1940's, high speed multiplications were measured in milliseconds and additions in tens and hundreds of microseconds. Today's proposed "high speed" computers boast of performing multiplication in a few microseconds and additions in a fraction of a microsecond.

Such increases in computing speeds have been achieved in numerous ways. Initial efforts toward higher computing speeds with existing components naturally led to parallel operation wherein entire words are processed simultaneously rather than bit by bit. Such tactics, of course, achieve higher speeds only at the expense of a large increase in circuit complexity and in the number of components used. Optimization of logical

design can also provide for a certain amount of increased speed, and at the same time, can lead to a reduction in the number of components required. Research in computer components has led to further increases in operating speeds. Development of memory devices such as the ferromagnetic core has provided the prospect of fairly high capacity storage coupled with rapid access. Steady improvement of transistors and the development of new concepts such as the surface barrier transistor and the drift transistor give the prospect of combining the high speeds heretofore restricted to vacuum tube circuits with the reliability and compactness of transistor circuits.

More recently higher computing speeds are being obtained by applying the concept of parallelism to an entire computing system. Several computers or computer systems are currently being developed which achieve greater overall computing speed by performing many complete logical operations simultaneously. Such techniques permit simultaneous operation of different sections of the computer or overlapping of certain unrelated operations. They allow more efficient use to be made of various elements in a system by eliminating much of the "dead time" which often results from coupling together units with different operating speed capabilities.

And still the demand for ever higher operating speeds remains. Led by various government research agencies, the search continues for new components and techniques which will lead to higher computing speeds. Typical is an ONR contract under which a Laboratory was requested to conduct "research leading to the development of components for the physical realization of digital computing equipment capable of operating at pulse repetition frequencies substantially in excess of those presently found feasible". With the continued improvements in vacuum tubes and

high speed transistors it is difficult to predict just how high operating speeds can be pushed using conventional circuit techniques. It appears, however, that repetition rate limitations are being encountered in the 10-100 mcs range. When the basic repetition or clock frequency is in the very high frequency range, much of the available power must be used to overcome the shunt loading of circuit and tube capacitances. Moreover, when the wavelength of the energy being used becomes significant compared to the physical dimensions of the circuit additional power is lost through radiation. Information signals must be isolated, reshaped, and retimed so often and at such high cost that increases in operating speeds are only obtained with prohibitive increases in circuit complexity, size, and power.

Optimization of logical design is of limited usefulness in achieving higher operating speeds, and parallel operation naturally has practical limitations since the point is eventually reached where cost and circuit complexity increase completely out of proportion to the improvement in operating speed.

Techniques involving multiplexing and carrying out of many complete logical operations simultaneously have much to recommend them, and in spite of their inevitable complexity probably offer one of the best solutions to the problem. It must be pointed out, however, that although two arithmetic units can certainly add twice as many pairs of numbers in any given time, if these two arithmetic units are operating on the same problem they will inevitably be feeding each other results, either directly or indirectly through a common storage device. Supervisory controls will therefore be required to prevent the two units from interfering with each other. Obviously then, more than twice as much equipment is needed to double the operating speed, not to mention the increased complexity of programming of the problem. Furthermore, because a large number of

mathematical and logical problems handled by computers are solved by essentially sequential processes, it would appear that parallelism as a means for gaining speed is at best a second choice. Though it does have its virtues, it is usually resorted to when no more speed can be obtained by other means.

It seems quite possible that in the speed range above that where conventional techniques are repetition-rate-limited, microwave circuits will find ready application. The wide bandwidth available at microwave frequencies is inherently capable of higher information-flow rates than are possible using the circuits now employed in electronic digital computers. For operation at repetition rates above 100 mcs., pulses of the order of 1 millimicrosecond (mμsec.) duration are required. The rise time, τ, of rectangular pulses is commonly related to the transmission bandwidth by $\tau = 1/2\ f_{bw}$. Therefore, if a rise time of 0.2 mμsec is desired, a bandwidth of $f_{bw} = 2500$ mcs would be required. Such a bandwidth cannot be obtained with conventional vacuum tube circuitry. Traveling-wave tube amplifiers on the other hand, offer the possibility of high gain over extremely broad bandwidths centered about a high r.f. frequency. At a frequency of 10 kmcs, a bandwidth of $\pm$ 2500 mcs is entirely within the state of the art. It would therefore seem desirable to investigate techniques by which short r.f. pulses of the order of 1 mμsec duration might be used in the arithmetic manipulation of numbers in a digital computer.

It will be the purpose of this paper to discuss various techniques of this nature which have been proposed. Various methods come to mind by which information may be stored in an r.f. pulse. The most obvious, of course, is that derived by direct analogy to lower frequency pulse circuits wherein the information content is manifest in the presence or

absence of a pulse. Other more sophisticated methods are those in which (a) the information of a pulse is contained in the frequency of the r.f. energy within the pulse, or (b) the information of a pulse is contained in the phase of the r.f. energy within the pulse. The manner in which an r.f. pulse is used to represent information will hereafter be referred to as the "script" of the particular system. Hence, the methods described above will be designated as "pulse-no pulse script", "frequency script", and "phase script", respectively.

Techniques for utilization of the pulse-no pulse script are discussed briefly in Chapter II. This treatment is necessarily brief because of the lack of any proposed gating circuits at the present time. It is included here primarily for the sake of completeness and because it aids in the logical transition from conventional computer techniques to microwave methods.

Frequency-domain techniques have been the object of considerable study during recent years. Much of this effort has been spurred by countermeasures applications where the frequency memory concept forms the heart of many proposed countermeasures systems. The use of a frequency script with r.f. pulses at microwave frequencies for computer applications is discussed in Chapter IV. Methods are discussed for representing, storing, and performing arithmetic and logical operations upon numerical information contained in the frequency of r.f. pulses. Multifrequency oscillators utilizing traveling wave tubes and delay lines are combined with fixed-frequency oscillators, radio-frequency pulse-controlled gates, balanced modulators, directional couplers, and related devices to form computing circuit arrangements. Such configurations are capable of performing arithmetic operations on numbers by operating upon groups of radio-frequency pulses with selected frequencies that represent

these numbers to form other groups of pulses with frequencies that represent the sums, difference, etc., of the input numbers. The use of a frequency script presents the possibility for using a radix greater than two since frequency memory devices can be made which exhibit stable operation at any one of a large number of selected frequencies. The prospect of operating in a decimal rather than a binary mode further increases the attractiveness of frequency domain techniques.

It is apparent that the apparatus necessary for the application of the frequency-domain technique, or for that matter, any microwave computer technique, will be complex and costly. The use of the phase script however, does permit certain logical operations to be accomplished by simple and relatively inexpensive devices, and in this respect it is perhaps the most promising of the microwave computer techniques. For this reason an extensive discussion of a phase script technique wherein a pulse of certain r.f. phase represents a binary "0" and a pulse of opposite phase represents a binary "1" is given in Chapter III. Devices have been proposed which are capable of generating, storing and manipulating such pulses. These devices are described and numerous logical circuits are visualized which are capable of accomplishing arithmetic operations by use of these techniques. Finally an attempt is made at combining these logical circuits to form an arithmetic-unit. The result is the logical description of an arithmetic-unit operating with a phase script which is theoretically capable of algebraic addition of two 20 bit numbers in less than 100 musec, and corresponding multiplication in about 1.6 usec.

Such circuits, of course, are many years from practical realization. This discussion however, does indicate the inherent capability of microwave techniques for achieving extremely high computing speeds. Furthermore, the speeds indicated above are achieved in strictly serial operation

whereas any approach to such computing speeds using present conventional techniques could be accomplished only by employing the ultimate in parallelism. Further development and refinement of microwave techniques might therefore be expected to yield even higher computing speeds than those visualized herein.

CHAPTER II

PULSE-NO PULSE TECHNIQUES

It is common in many present high speed electronic digital computers to represent binary information by use of a pulse-no pulse script, that is, where the presence of a d.c. pulse represents a binary 1 and the absence of such a pulse represents a binary 0. As one attempts to extend computer operating speed by increasing the repetition rate, the pulses become narrower and the need is immediately apparent for broader band, high gain amplifiers to amplify and reshape these pulses after they have been attenuated and broadened in passing through gating elements. At a repetition rate of say 500 mc, which would require pulse widths of the order of 1 μsec, substantial amplification over a bandwidth of at least 2000 mcs is desired. Since adequate gain-bandwidth products are not available with conventional vacuum tube circuits for such narrow pulses one is led naturally to a consideration of the traveling wave tube amplifier* for which a gain of 30 db over a 2000 mcs band is not uncommon. It is necessary, however, to realize that in a traveling wave tube such bandwidths are centered about a high r.f. frequency, and hence satisfactory amplification of video pulses with a large d.c. component is not possible. Conventional gating elements, however, are conveniently designed to function with d.c. pulses, and hence are not directly compatible with travel-wave tube amplifiers. One possible solution to this problem is illustrated in Fig. 1.

*See Appendix II for a discussion of the Traveling Wave Tube Amplifier as a computer component.

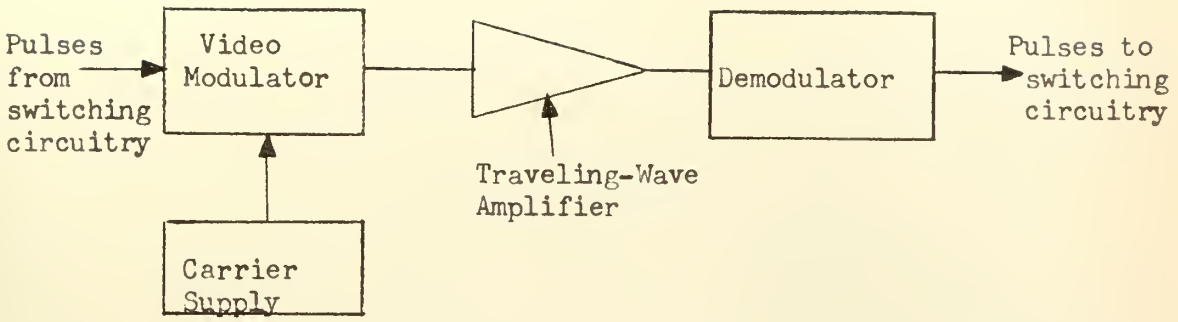


Fig. 1 - Possible Arrangement for Using Traveling Wave Amplifier in Conjunction with Conventional Gating Elements

Here gating elements of conventional types are used, and the pulses are used to modulate an r.f. carrier before going on the traveling wave amplifiers and demodulated after coming off. This procedure eliminates the compatibility problem mentioned above, but introduces numerous other troublesome factors such as the necessity of supplying carrier frequency, and the complication, losses, and delay involved in the modulation and demodulation processes.

An alternative solution would be the use of r.f. pulses rather than "d-c" pulses throughout the switching circuitry. This, of course, would require the development of logical gates which would respond to short pulse of energy at microwave frequencies. There are very few references to such logic elements in current literature, and this writer could find no description of any device which had actually been constructed which was capable of performing logical operations with millimicrosecond pulses at microwave frequencies.

It is not difficult, however, to visualize some microwave gating elements which might be constructed using such microwave components as traveling wave tubes and hybrid-T junctions.

For example, Figure 2 shows how a magic-T alone might be used as a logical element. If inputs "A" and "B" are identical r.f. pulses then no

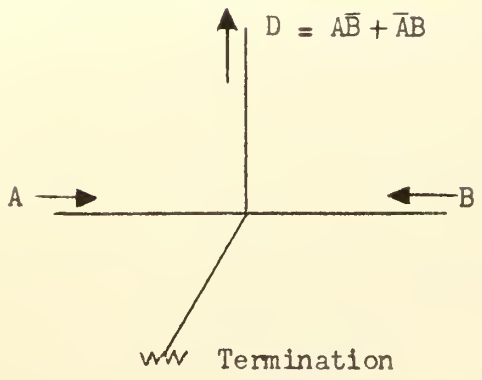


Figure 2 - Microwave "Exclusive OR" Circuit

output will occur at D, the difference arm. Thus the magic-T reacts as an "Exclusive OR" circuit producing an output pulse only when a pulse appears at "A" and not at "B", or when a pulse appears at "B" and not at "A". If "A" is now considered to be a clock pulse, present at all times, then this magic-T performs logical inversion as shown in Figure 3.

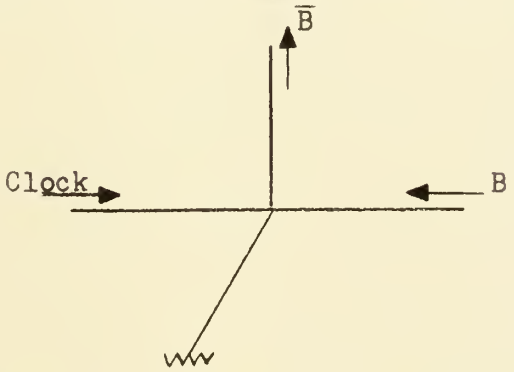


Figure 3 - Microwave Logical Inverter for Pulse-No Pulse Script

In this case, a pulse will appear at the output "D" only when no pulse appears at input "B".

Figure 4 illustrates how a magic-T might be used in conjunction with an amplitude limiter to produce a microwave "OR" gate. In this case the output "C" is from the sum arm of the magic-T and will give an output pulse when input pulses occur at either "A" or "B".

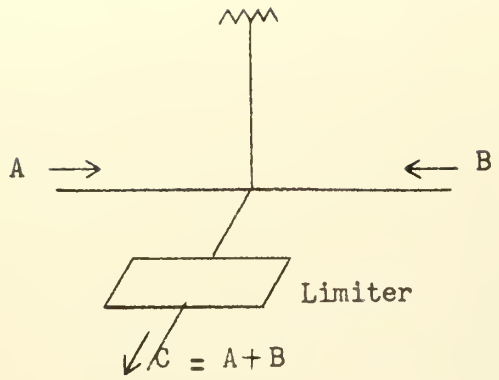


Figure 4 - Microwave "OR" Gate for Pulse-No Pulse Script

When input occurs at both "A" and "B", however, the two signals combine in the sum arm to give a signal of greater amplitude than when only one input was present. Hence a limiter is required so as to insure that the output amplitude is constant for the different input conditions.

In addition to these gates made from passive elements, the saturation effect of a traveling wave tube amplifier provides the means for constructing logical gates from an active element, and thus providing amplification as well as gating. Figure 5 is a sketch of a typical input-output characteristic of a traveling wave tube. As the input power increases beyond the point corresponding to the "knee" in the saturation curve the output power increases only slightly. Thus a traveling wave tube has the non-linear properties desirable for microwave gating.

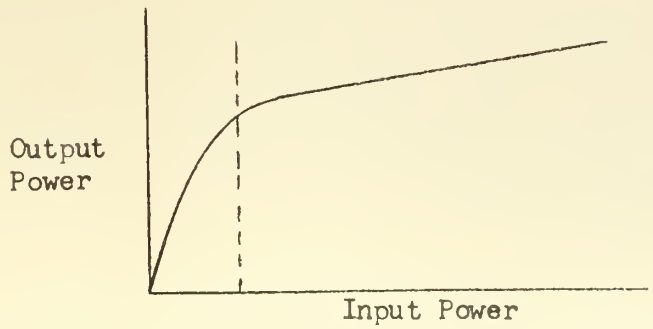
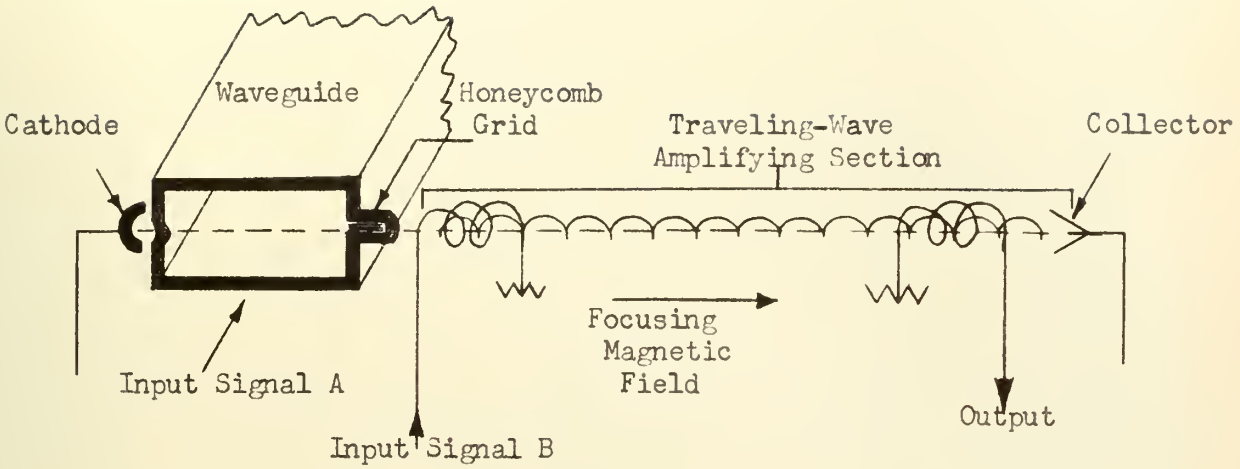
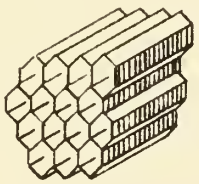


Figure 5 - Typical Traveling Wave Amplifier Input-Output Characteristic

It should be noted at this point that although the microwave logical elements discussed above respond to a pulse-no pulse script, they are critically phase dependent for proper operation. That is, satisfactory functioning of these gates requires a very precise relationship between the phases of the r.f. energy within the input signal pulses. Hence, a certain similarity will be noted between these gating elements and those to be described in the following chapter which respond to a phase script. It would appear to be desirable to have gating elements which respond to a pulse-no pulse script and are entirely independent of the r.f. phase of the input signals. One such device has been proposed by M.P. Forrer[1] of the General Electric Microwave Laboratory and is illustrated in Figure 6. This device is described as an "inhibit-gate" and functions such that the presence of a control signal A inhibits the transmission of a second signal B. This gating action is obtained by making use of the cyclotron resonance principle. The device consists of a conventional traveling wave tube amplifier with a section of wave guide inserted between the cathode and the helix in such a manner that the electron beam must pass across the wave guide before entering the helix. On the far side of the wave guide, away from the cathode and in the path of the electron beam is a honeycomb grid constructed as shown in Figure 6(b). Input signal B is introduced onto the helix of the traveling wave tube in the conventional manner while signal A is fed into the wave guide as illustrated. In the absence of signal A, signal B will be amplified in the normal manner. When signal A is applied, the combination of forces from its transverse electric fields and the magnetic focusing field will cause the electrons of the beam to spiral about the beam axis at a rate given by the cyclotron frequency $\omega = e/m\ B_x$. By proper adjustment of B_x, the magnetic flux, the cyclotron frequency may be made to coincide with the frequency of



(a)



Honeycomb Grid

(b)

Fig. 6 - Gated Traveling-Wave Tube using a Honeycomb Grid

signal A and a resonance condition will occur, i.e., the radius of spiraling increases as the electrons move across the wave guide. Thus, as the spiraling electrons pass through the honeycomb grid they intercept with the grid walls and the electron beam is cut off, thus suppressing transmission of signal B. Another interpretation of the logic performed by this device is noted if signal B is considered as a clock pulse. Then the device performs logical inversion of signal A, while at the same time giving an amplified and reshaped output.

It can easily be shown that any logical element may be constructed by properly combining logical inversion gates and logical "OR" gates. Thus, in principle the devices described above could be used to provide all the logic necessary for a microwave computer. Much more than logic elements, however, is needed to perform even the most basic functions of a computer, not the least of which are pulse generators and storage devices. Such devices are discussed in the next chapter in connection with the "phase script technique" as are some of the problems which might be encountered in attempting to assemble these elements into a unit capable of performing arithmetic operations. It will be noted that logic circuits which use a phase script are discussed considerably more extensively than those which use a pulse-no pulse script. However, most of the discussion would apply equally well to the "pulse-no pulse" circuits if the corresponding gating elements were used.

The use of microwave elements in digital computers is still very much in the early stages of development, and hence, any commercial companies investigating such techniques consider their work to be highly proprietary. Very little has been published on the subject and hence it is difficult to know just what approaches are being made to the problem. This author has chosen the phase script as a vehicle for discussion of

microwave techniques in computers because of the availability to him of information concerning various proposed logical gates, pulse generators and storage devices through his association with the General Electric Computer Laboratory, Palo Alto, California.

CHAPTER III

PHASE SCRIPT TECHNIQUES

In the preceding discussion of microwave gates operating with a pulse-no pulse script it was mentioned that if such gates were constructed using magic-T's as components, strict control of the phase of the input pulses was required for satisfactory operation of the gate. Phase script techniques recognize this requirement for phase control of the pulses and go one step further in that the phase of the pulse is used to represent the information content of the pulse. In the discussion throughout this chapter a "phase script" will be assumed wherein a binary "zero" is represented by a pulse of a given r.f. phase and a binary "one" is represented by a pulse of opposite r.f. phase. For illustrative convenience a schematic notation for these pulses will be used as shown in Fig. 7. It should be noted that although in the schematic notation pulses are represented by a single cycle, actually the pulses will contain many cycles of r.f. energy. For instance, a 1 μsec pulse at 10 kmc would contain 10 cycles of the 10 kmc signal.

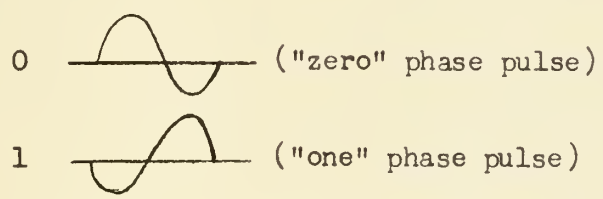


Figure 7 - Definition and Schematic Notation of Phase Script

The attractiveness of a phase script for use in a microwave computer arises from the relative simplicity of the proposed gating structures (to be described below) and in the ease with which negation can be accomplished, i.e., a 180° phase shift of a signal at any point merely requires a change in the transmission path length by $\frac{\lambda}{2}$ of the signal frequency.

3.1 Phase Script Gates

A device which operates with a phase script and is capable of simultaneously giving outputs which represent the binary functions $A \cdot B$, $A+B$, $A \cdot \overline{B}$, and $A+\overline{B}$* has been proposed by M.P. Forrer[2] of the General Electric Microwave Laboratory at Palo Alto. Such a device is illustrated by Fig. 8 and its operation is characterized by the table shown in Fig. 9.

The circuit consists of three broadband magic-T's and four amplitude limiters connected as shown. The sum and difference arms of the three T's are indicated by S_1, D_1; S_2, D_2; and S_3, D_3, respectively. The "AND" and "OR" outputs of the circuit at S_2 and D_2, respectively, for the four possible input combinations are illustrated by Fig. 9. The $A \cdot \overline{B}$ and $A+\overline{B}$ outputs at S_3 and D_3, respectively, can be readily verified by setting up a similar table. The need for an amplitude limiter at each of the various outputs is apparent from Fig. 9 if these outputs are to drive other logical circuits of similar types, since operation of this device is based upon equal amplitude inputs.

It should be noted that any one or any desired combination of the four possible outputs can be obtained merely by using only the required number of magic-T's and placing terminations at the unused outputs. This single device then can be used to provide all the logic which would be required in a digital computer.

*See Appendix I for explanation of binary function symbols.

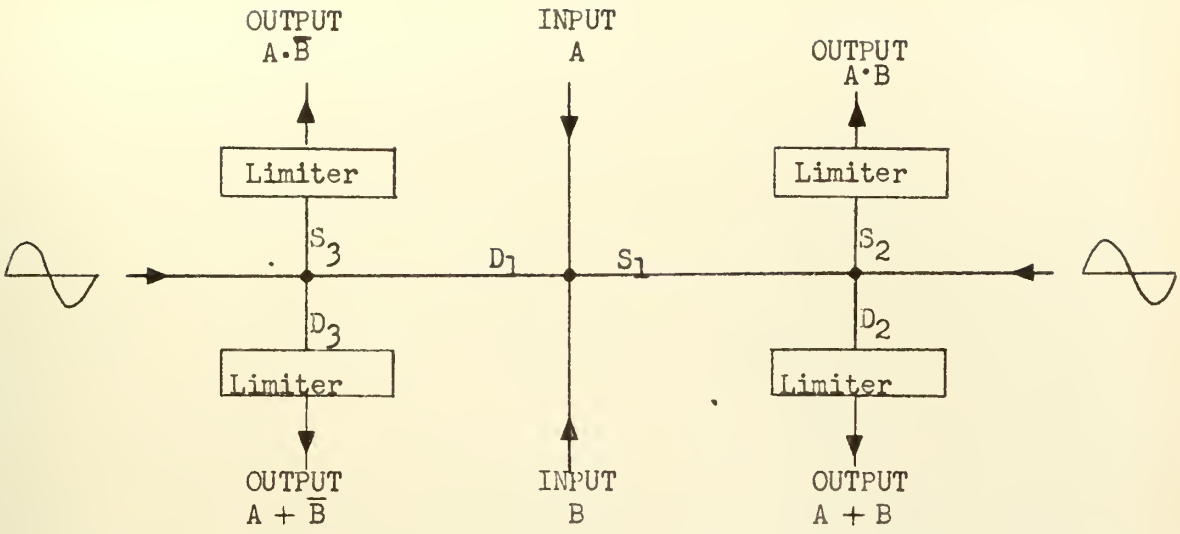


Fig. 8 - Logical circuit producing simultaneously the four functions $A + B$, $A \cdot B$, $A + \bar{B}$, $A \cdot \bar{B}$

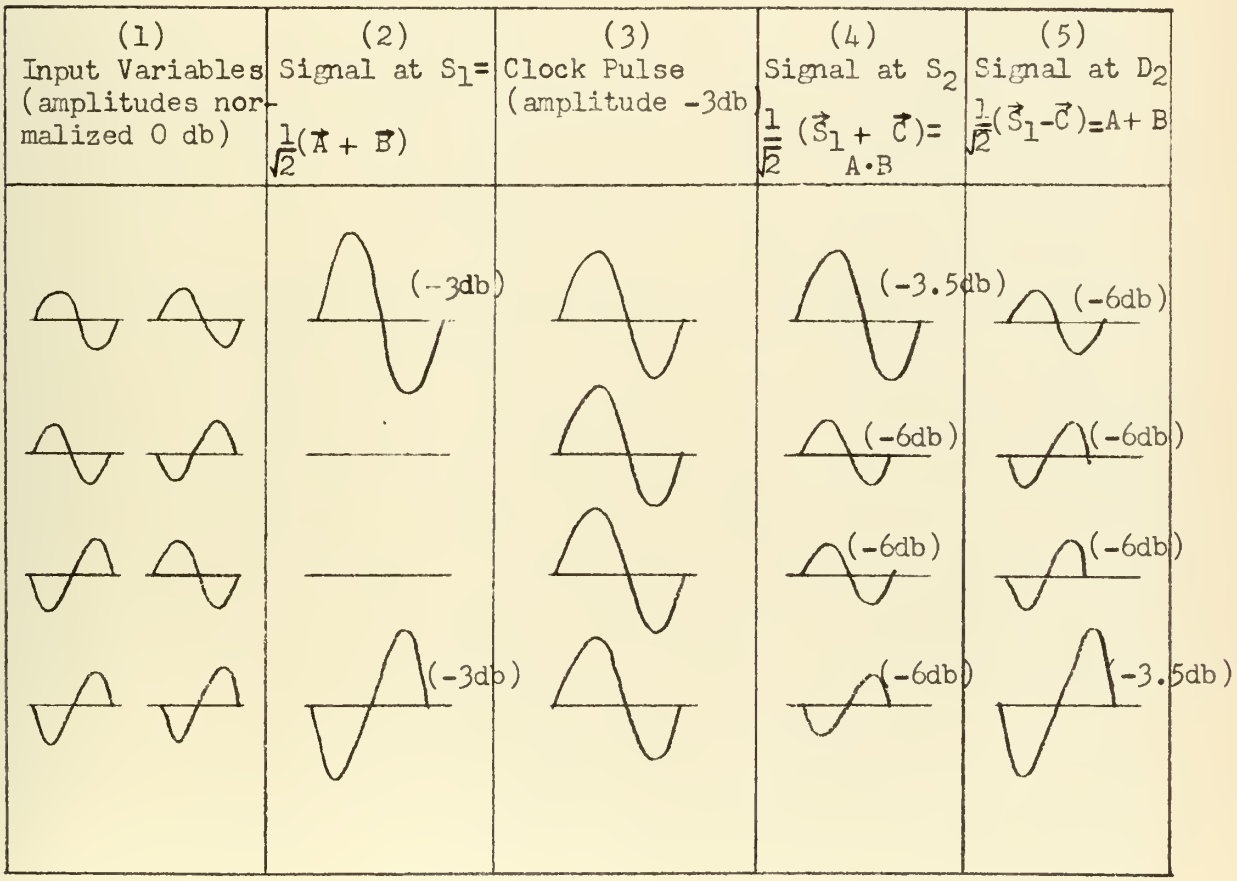


Fig. 9 - Phase and amplitude relations of all possible signal combinations in the AND and OR circuits of Figure 8.

3.2 Regenerative Memory Unit

A Regenerative Memory Unit for use in a phase domain computer has been proposed by Dr. W. A. Edson [3]. The essential fact which underlies the operation of the system is that any frequency-halving device represents a kind of binary memory unit since the phase of the output may be either positive or negative with respect to a given cycle of the input frequency. One proposed system is shown in Figure 10.

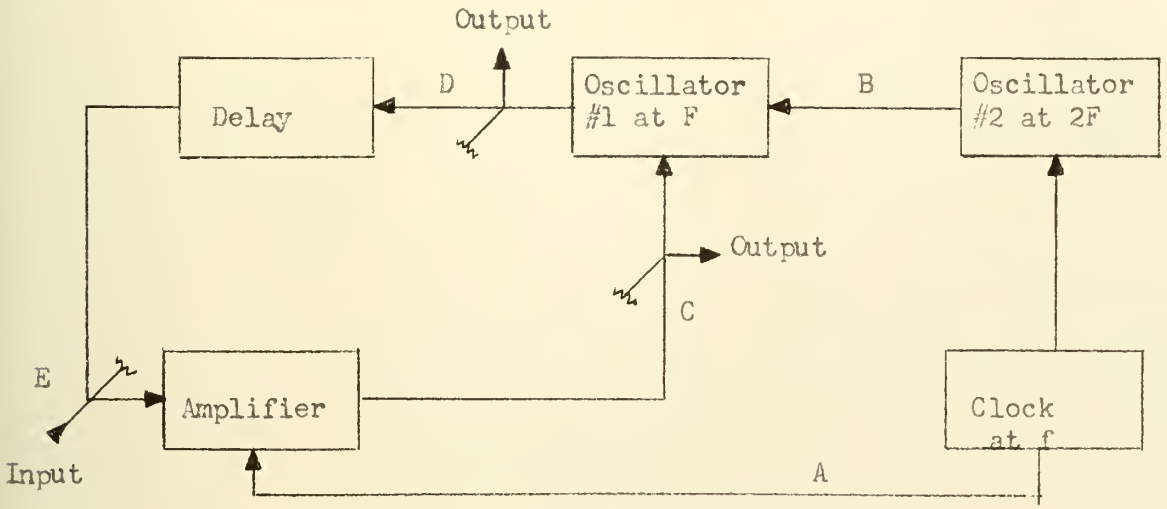


Fig. 10 - Block Diagram of a Regenerative Recirculating Memory Unit for Microwave Phase-Script Computer

The regenerative memory loop is composed of the amplifier, oscillator #1 at F and the delay. Information is stored in the loop in the form of N bits where each bit is represented by an r.f. pulse of either positive or negative phase. The number of bits which can be stored in the loop is determined by the total loop delay. The timing and phase of the pulses is controlled by the clock at F and Oscillator #2 at 2F.

Oscillator #1 is driven by Oscillator #2 at 2F and operates as a frequency-halving device at F. The phase of the output of oscillator #1 is controlled by the phase of the input pulses at C. As successive

pulses arrive at C the output at D varies in phase accordingly. The output of Oscillator #1 feeds to the amplifier where the clock causes the gain of the amplifier to be varied at frequency f. The output of the amplifier then contains the reshaped and retimed pulses of positive and negative phase which are fed to Oscillator #1.

Information can be read into the loop from a separate source at E in the form of pulses of positive or negative phase. Information can be read in through a hybrid junction as illustrated or by a directional coupler. Output can be taken from either C or D by means of a hybrid junction or a directional coupler, and if taken at C would consist of a train of pulses of positive or negative phase.

Under typical operating conditions the train of N bits of information repeatedly circulates around the closed loop. The output of Oscillator #1 is continuous, but the phases reverse at intervals which are multiples of the clock period.

3.3 Phase Script Logic Design Considerations

Means for accomplishing logical gating and memory, the two basic functions required of any computing circuit, have been described above. It is now possible to visualize means by which such devices may be combined into logical circuitry which would perform arithmetic operations. There are, however, numerous properties of these devices which must be kept constantly in mind when designing logical circuits which make the problem somewhat different than that of logical design using conventional diode-resistor gates.

First of all, when using the phase script with which we were primarily concerned because of the relative simplicity of the "AND" and "OR" gates and of the operation of negation, it must be noted that a 6 db reduction in signal level is experienced through each gate. This property severely

limits the number of logic levels which may be employed before amplification of the signal is required. Assuming 30 db as a nominal value for gain available from a traveling wave tube amplifier, it can be seen that a maximum of five gates may be used in series before amplification is required.

This then leads to a consideration of the delay involved in a traveling wave tube amplifier which is relatively long compared to the 2 μsec or less pulse period which we desire to operate. Attempts must be made to insert these amplifiers at such points in the circuitry where the delay will be least likely to slow down the overall computing speed.

Another limitation can be found in the fact that only two inputs can be allowed to any gate. This, of course, presents no fundamental limitation in that a multiplicity of inputs can be handled merely by cascading gates. However, in view of the 6 db signal loss through each gate combined with the delay required for TWT amplification, this practice is of limited usefulness. Care must also be taken to insure that the two inputs are of equal amplitude, since successful operation of the gates is dependent upon this fact.

Timing, of course, is an important consideration in such a computer, because of the extremely short pulses and the high repetition rate. Whereas, in a conventional computer pulse widths are such that a clock pulse may be considered to appear practically instantaneously at all points throughout the computer, such is not the case when pulses as short as 1 μsec are employed. In addition, the use of phase script requires strict attention to transmission path length since the difference between a "0" and a "1" is only one-half wavelength of the r.f. energy in an ideal transmission line.

3.4 Phase Script Adder Design

Conventional logic for the addition of two binary numbers is readily derived from the binary truth table for such an operation which is shown in Fig. 11. In this table A and B represent corresponding orders of the two numbers to be added while C represents the carry digit from the next lower order. The sum is represented by S, and C' is the carry digit which results from the sum of A, B, and C, and which will be passed on to become C for the next higher order addition. This table represents all possible combinations of the three input variables and the corresponding sum and carry values.

The Boolean expressions for the sum and carry are readily written:

$$\text{Sum} = (\overline{A}\cdot\overline{B}\cdot C)+(\overline{A}\cdot B\cdot\overline{C})+(A\cdot B\cdot C)+(A\cdot\overline{B}\cdot\overline{C})$$

$$\text{Carry} = (\overline{A}\cdot B\cdot C)+(A\cdot\overline{B}\cdot C)+(A\cdot B\cdot\overline{C})+(A\cdot B\cdot C)$$

By algebraic manipulation and through use of Boolean Algebra identities, these expressions may be written in any number of forms. One form which seems particularly appropriate for our use, keeping in mind the ease with which negation may be accomplished with a phase script, is given below.

$$\text{Sum} = C(A\cdot B+\overline{A}\cdot\overline{B})+\overline{C}(\overline{A}\cdot B+A\cdot\overline{B})$$

$$\text{Carry} = A\cdot B+C(A+B)$$

Making use of the Boolean algebra identities

$$A\cdot B+\overline{(A+B)} = A\cdot B+\overline{A}\cdot\overline{B} \quad \text{and} \quad \overline{(A\cdot B+\overline{A}\cdot\overline{B})} = \overline{A}\cdot B+A\cdot\overline{B}$$

an adder representing the Boolean equations for sum and carry shown above can be realized as shown in Figure 13. This circuit uses four logic levels to generate the sum and three logic levels to generate the carry. The carry from the addition of the two next lower order digits is fed back into the second level of the carry chain and the third level of the sum chain.

The symbols used to represent the various phase script logic elements are described in Figure 12. It should be noted that any number from one to four of the available outputs of the phase script gate can be used to form a separate gate. It should also be noted as indicated in Figure 12(b) that for the operations indicated by the symbols "$\underline{+}$" and "$\underline{\cdot}$", the negation goes with the lower of the two inputs. Figure 12(d) represents an ideal amplifier of no delay, with gain indicated in db.

Although the logical design of this circuit appears rather straightforward and simple to attain, it is necessary to analyse it critically from the standpoint of timing and gain required to realize its practical limitations.

A	B	C	S Sum	C' Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Fig. 11 - Truth Table for Binary Addition

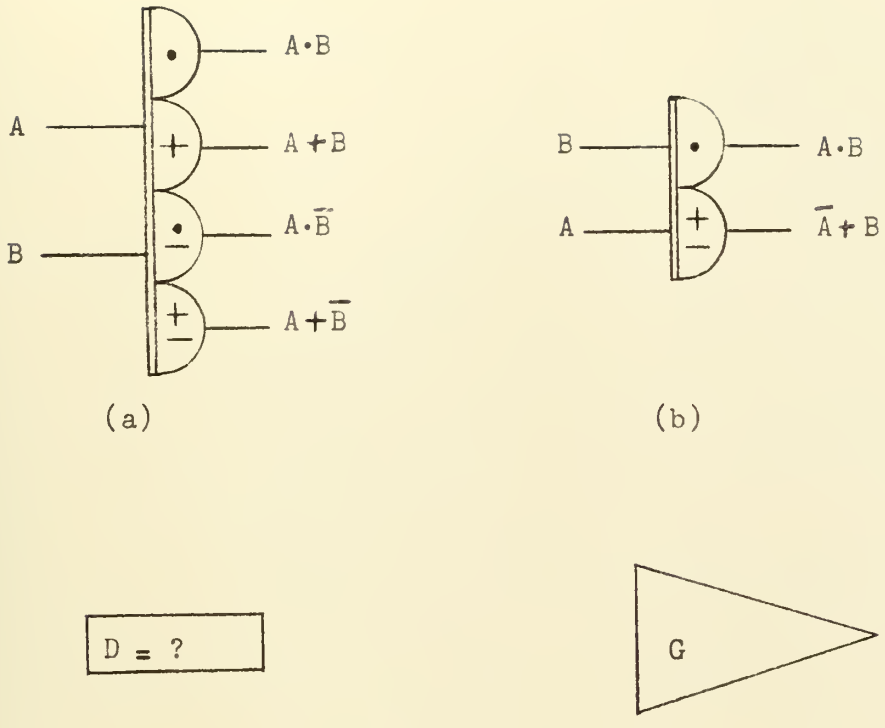


Fig. 12 - Symbols used in Phase Script Logic Circuits

Consider the input signals normalized to 0 db. Then with a 6 db reduction in signal through each logic level and realizing that a 3 db reduction occurs if the signal is aplit into two parts, the relative signal level at various points throughout the adder can be determined and is indicated by the small numbers at the input and output of each gate. It can be easily seen that since equal amplitude inputs are required for the gates a minimum of 12 db amplification of the carry signal is required if it is to be inserted at the second logic level in the carry formation chain. An additional 1 db is required to allow for its insertion into the third logic level of the sum formation chain. Thus a total of at least 13 db amplification is required for the carry return, neglecting all other transmission losses. Furthermore, the carry must be inserted at precisely the right instant to coincide with the pulses representing the next higher order digits.

Determination of the timing relationships in the adder of Fig. 13 is facilitated by examination of Fig. 14. On this chart, time is measured from left to right with zero time corresponding to the instant in which the leading edge of the first pulse enters the first logic level. Each of the four logic levels is represented by a separate horizontal line. It is assumed that propagation of a pulse through the gating elements requires 0.4 musec. The shaded areas represent the first pair of 1 musec pulses. The time during which these pulses enter a given logic level is indicated by the length of the shaded area. The second pair of pulses is represented by the cross-hatched area. The solid lines indicate the propagation of the pulses through the sum generating chain while the dashed lines show the carry formation.

$$\text{Sum} = C(A \cdot B + \overline{A} \cdot \overline{B}) + \overline{C}(A \cdot \overline{B} + \overline{A} \cdot B)$$

$$\text{Carry} = A \cdot B + C(A + B)$$

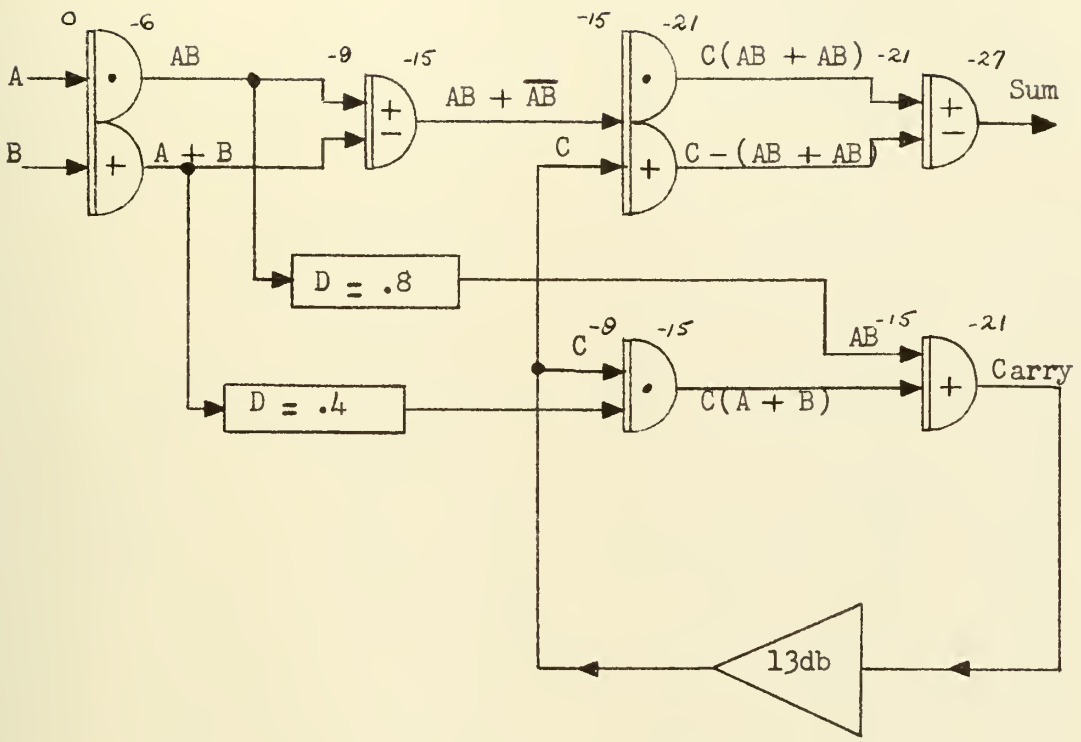


Fig. 13 - Logic Diagram for Phase Script Adder

It can be seen from Fig. 14 that a total time of 1.6 mμsec is required to generate the carry signal from the first pair of pulses. This carry signal must be reinserted at the third logical level to coincide with the second pair of pulses. These pulses reach the third logic level at time 2.8 mμsec and hence the carry pulse must be delayed 1.2 mμsec before being reinserted.

It is now apparent that in order to obtain satisfactory operation from the adder circuit of Fig. 13, a traveling wave tube amplifier is required which will give a gain of at least 13 db with a delay of no more than 1.2 mμsec!

Such a traveling wave tube amplifier is not in existence today to the knowledge of this writer. It was therefore felt desirable at this point to investigate the properties of traveling wave tubes to determine whether there were any fundamental limitations which would prohibit the development of such a tube. The results of this investigation are included in Appendix I, a discussion of the traveling wave tube as a computer component. This investigation revealed no apparent reasons why such a tube could not be built, however it also pointed out that present commercially available traveling wave tubes are far from realizing the theoretical values of gain with short delay.

An alternate adder circuit is shown in Fig. 15. In this circuit two half adders are combined in series to form a full adder. The truth table and Boolean equations for a half adder shown in Fig. 18 show that a half adder considers only the partial sum of the addition of two binary digits, neglecting the effects of a carry from the preceding order. Thus, it is necessary to follow the half adder by another half adder in which the partial sum is added to the carry from the preceding order. A carry produced in the first half adder is delayed and joined in an "OR" gate with

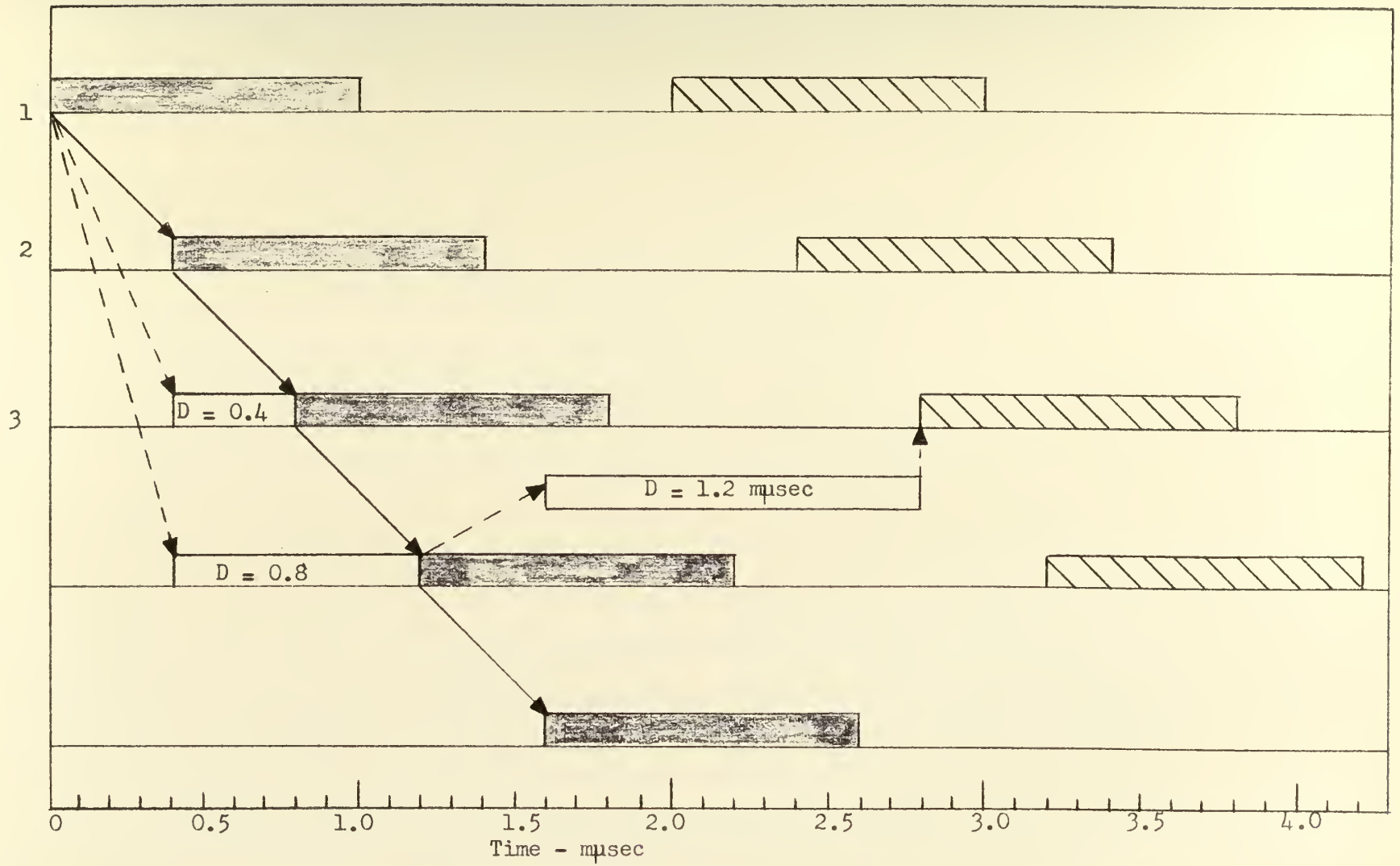


Fig. 14 - Timing Chart for Phase Script Adder of Figure

a carry produced in the second half adder to form the new carry for the next higher order digits.

The relative power levels shown in Figure 15 indicate that this circuit requires only 12 db gain in the carry feedback circuit, a saving of 1 db over the circuit of Figure 13. This is so because the carry is only required to feed one gate instead of two. The timing chart in Figure 17 shows that the same delay is required for the carry feedback loop as was required in the circuit of Figure 13. An additional advantage to the circuit of Figure 15 is to be found in the signal level of the sum digit which is -24 db as compared to -27 db in the previous circuit. This saving in db is made possible by making full use of the simultaneous outputs available from the phase script gates, thereby obviating the necessity of splitting any of the outputs and experiencing the attendant 3 db loss.

Net add time in the two circuits just described would be approximately one word time plus the time required to propagate through the adder. Assuming a 20 bit word length, 2 μsec per bit, and 0.4 μsec propagation time per logic level, net add time is about 42 μsec.

Both of the adder circuits described so far require a short delay, high gain traveling wave tube amplifier for successful operation. It was felt desirable to investigate other means for accomplishing addition without this requirement. One possible method is shown in Fig. 18. This circuit is merely a combination of half adders in series which form successive partial sums with the delayed carry from the next lower order partial sum. This scheme, of course, would require a number of half adders equal to the number of bits being processed plus two conventional traveling wave tube amplifiers for each three half adders. The principle

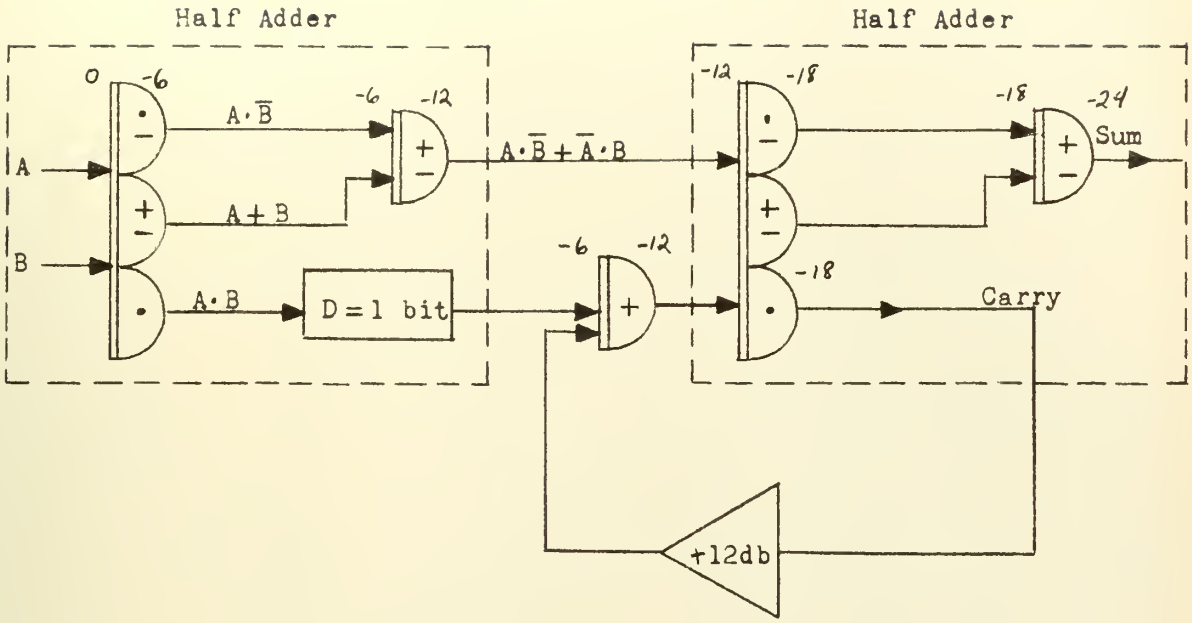


Figure - 15 Logic Diagram for Phase Script Adder

A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Sum $A\cdot\overline{B}+\overline{A}\cdot B$

Carry $A\cdot B$

Figure 16 - Truth Table and Boolean Equations for Half Adder

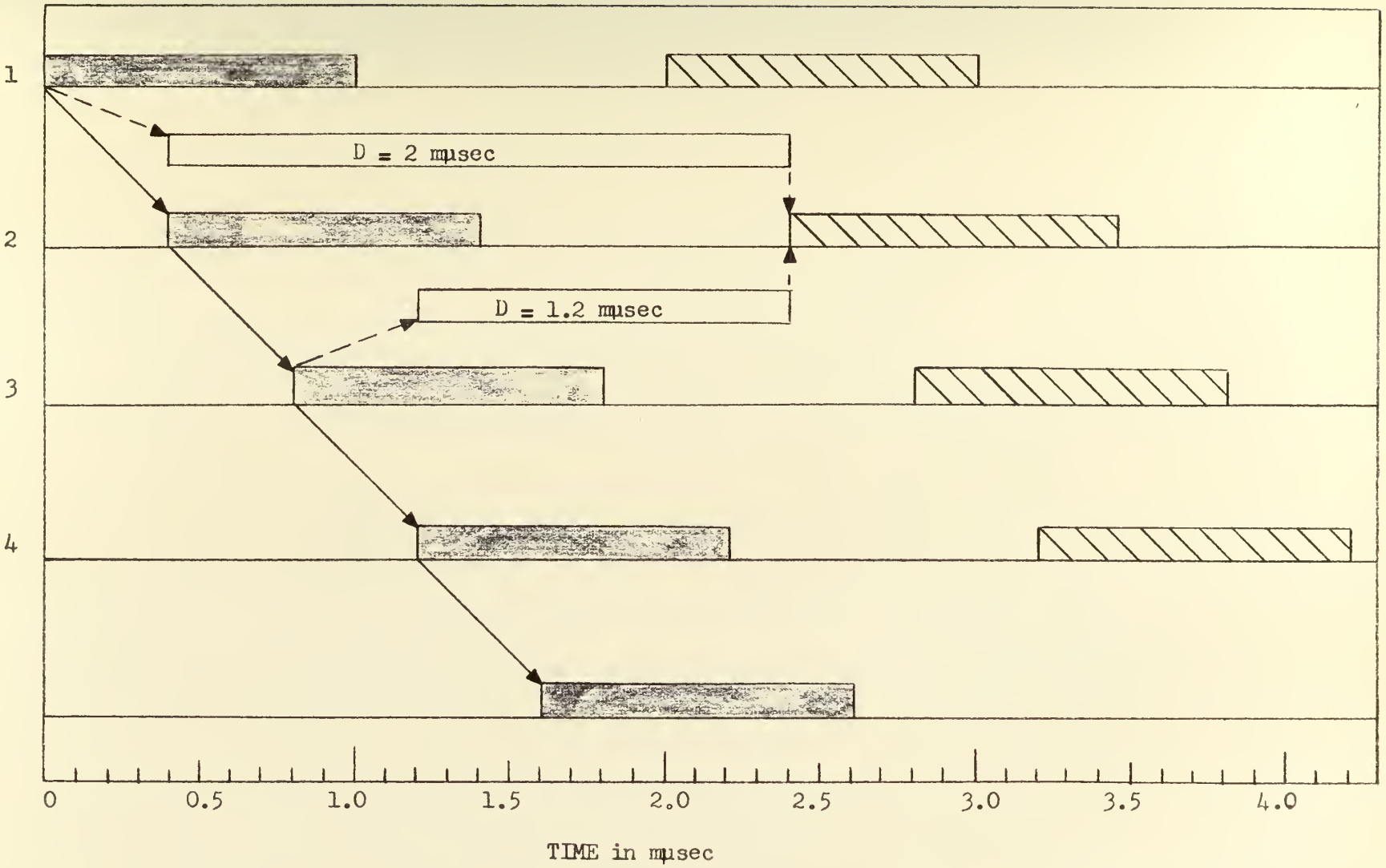


Fig. 17 - Timing Chart for Phase Script Adder of Figure

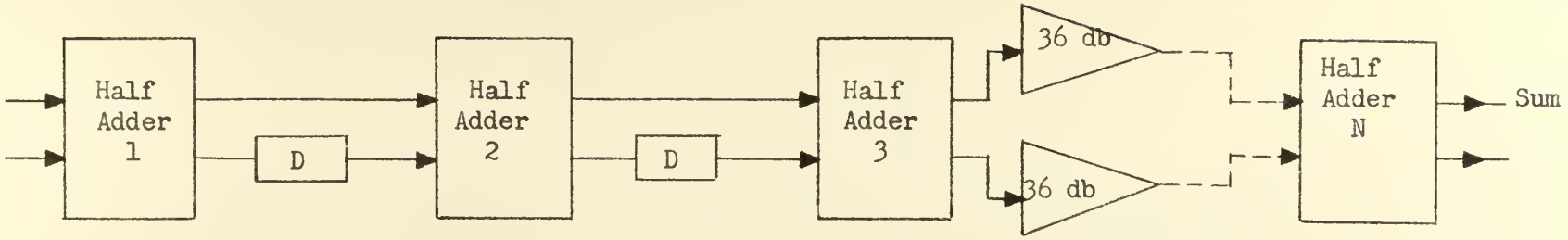


Fig. 18 - Block Diagram of Binary Adder

of this method can be illustrated by a simple example of the addition of two four bit numbers. Consider the addition of the binary numbers 0111 and 0001:

	0111	
1st Half Adder	0001	
	0110	S_1
2nd Half Adder	001	C_1
	0100	S_2
3rd Half Adder	01	C_2
	0000	S_3
4th Half Adder	1	C_3
	1000	S_4

Net add time in such a circuit would be approximately one word time plus propagation time. For a 30 bit word

1 word time	40 musec
20 half adders at 1.0 musec ea.	20 musec
10 TWA at 10 musec ea.	100 musec
	160 musec

and the total equipment required would be 210 magic T's and 20 traveling wave tube amplifiers. The large amount of expensive equipment, of course, makes this method for addition rather impractical. It does, however, have the advantage that the repetition rate could be increased without having to worry about carry return time. In the two circuits first described, repetition rates much above 500 mcs are pretty much out of the question since this would require carry return and amplification in less than 1 musec.

3.5 Phase Script Subtractor

The adder circuit of Fig. 13 may be readily modified by conventional logic design techniques to provide for the alternate operation of subtraction. This is done by merely introducing another logic channel for generation of a borrow signal and a switch to feedback the borrow signal instead of the carry when subtraction is desired. Subtraction however, is more readily accomplished by addition of complements, a procedure which will be discussed

more thoroughly in connection with the overall design of an arithmetic unit. Because of its similarity to the adder circuit of Fig. 13, no further discussion of the subtractor circuit will be given here.

PHASE SCRIPT MULTIPLIER

Binary multiplication and methods for accomplishing it in digital computers have been extensively discussed by R.K. Richards [10]. The means by which binary multiplication is accomplished is best illustrated by a numerical example:

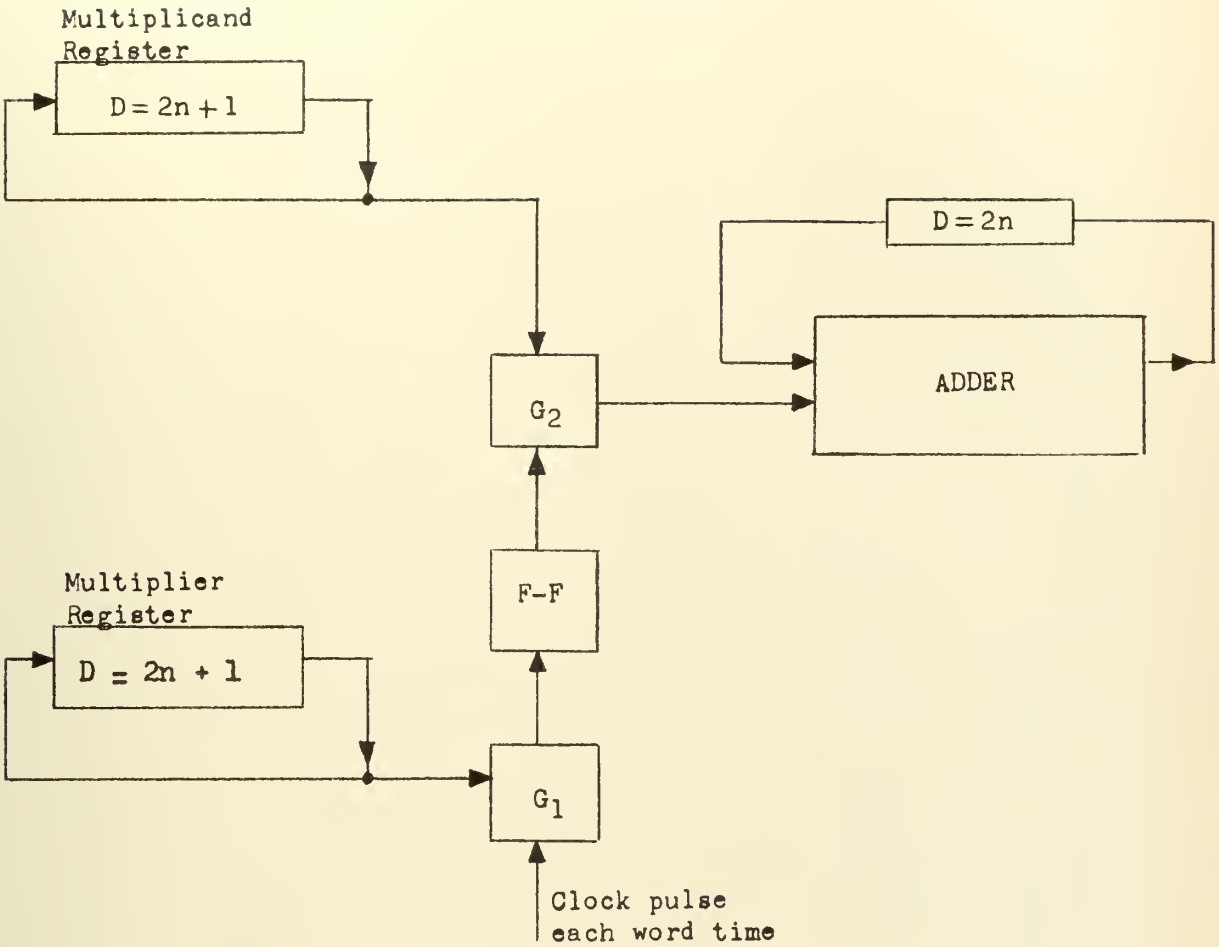
	Multiplicand	1111
	Multiplier	1101
	A	1111
Partial Products	B	0000
	C	1111
	D	1111
Product		11000011

In the foregoing example of a binary multiplication, 1111 (decimal 15) is multiplied by 1101 (decimal 13) to obtain the product 11000011 (decimal 195). The partial products are clearly zero or equal to the multiplicand, according to whether the corresponding multiplier digit is 0 or 1. That the partial products are recorded in the proper columns (orders) should be apparent to anyone who is at all familiar with multiplication procedure. The customary way to sum the partial products is to add the digits in the partial products, one column at a time, starting with the lowest order. In a computer, however, it is generally much simpler to add the complete partial products one at a time in the formation of the product. Thus we may take partial product A and add it to partial product B to form a first partial sum. Partial product C is then added to this partial sum to give a new partial sum, and finally, partial product D is added to the latest partial sum to give the final product. It should now be apparent that the process of binary multiplication is

nothing more than successive additions of the multiplicand to itself depending upon whether the corresponding multiplier digit is zero (do not add) or one (add), with each addition accompanied by a shift of the multiplicand one order to the left.

A functional block diagram of a serial binary multiplier which operates in this manner is shown in Fig. 19. The multiplier and multiplicand consists of a recirculating loop in which pulses are circulated in serial fashion. The total delay of the multiplier loop is 2n-1 pulse times and the total delay of the multiplicand loop is 2n+1 pulse times, where n is the number of binary digits in each operand. As the two lowest order digits emerge from the storage registers a clock pulse is applied to gate G_1. If the multiplier digit is a 1, it sets FF to 1, which in turn opens Gate G_2, which passes all the digits of the multiplicand into the adder. At the beginning of the next word time (2n pulse times later) the second order digit of the multiplier is at G_1. This is so because the delay of the multiplier register loop is 2n-1 pulse times, and therefore, on each successive word time the next higher order multiplier digit will appear at G_1. When the clock pulse opens G_1, FF is set, and if 1, the multiplicand is again added to the contents of the adder loop. Now, however, since the delay of the multiplicand register loop is one bit time greater than the delay of the adder loop, the multiplicand will be shifted one bit to the left before it is fed into the adder. In this manner the process is repeated, the multiplicand shifting one place for each circulation and being added to the partial product each time a 1 appears in consecutive higher order positions of the multiplier.

In the implementation of a multiplier circuit using the microwave devices previously described, we were now faced with the problem of how to realize the function represented by the block labeled FF in Fig. 19.



n = number of digits in operand

word time = 2n pulse times

Figure 19 - Functional Block Diagram of a Serial Binary Multiplier

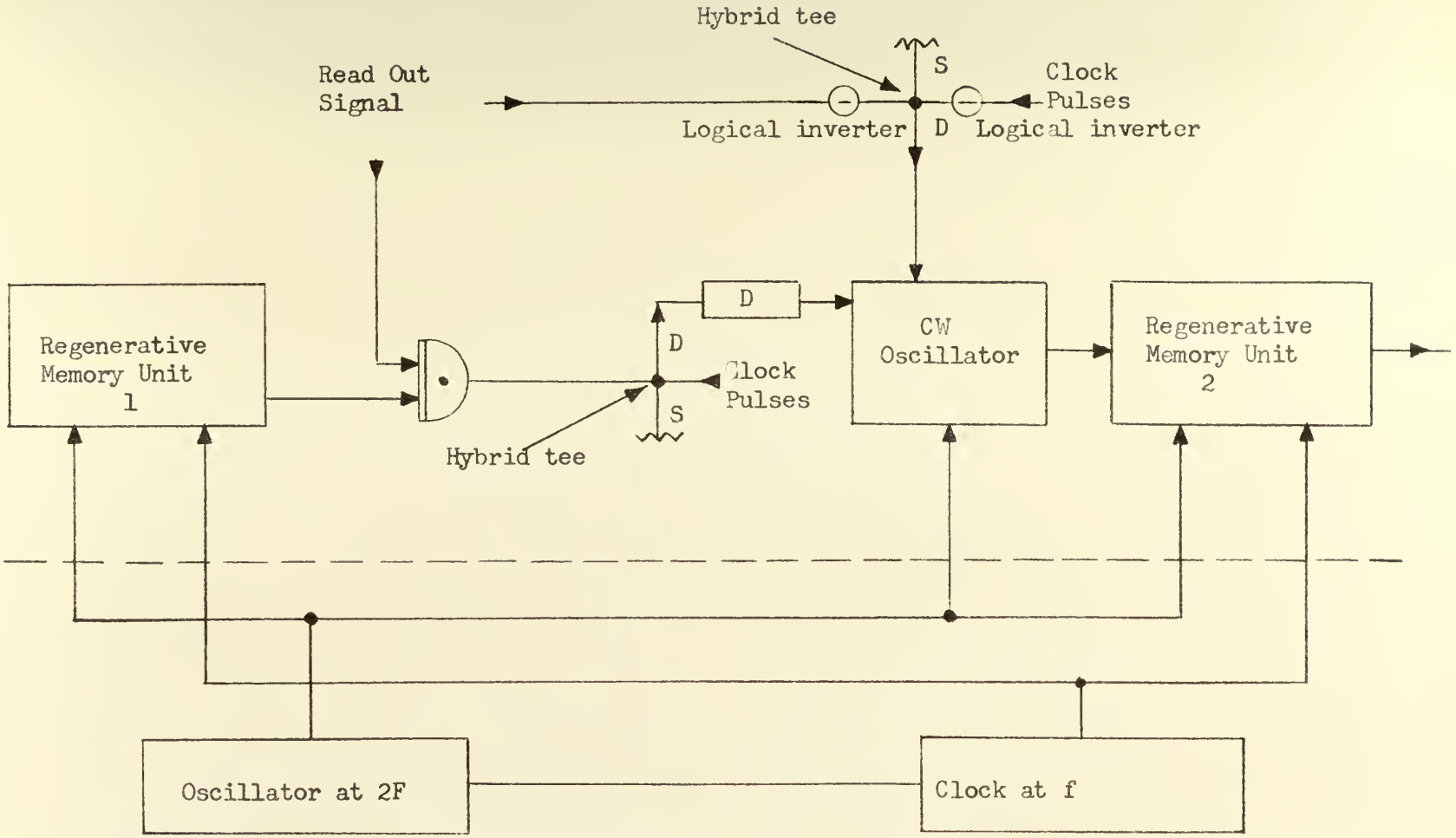


Fig. 20 - Single Bit Read Out System

The requirements for such a device are that it provide a means for selecting one particular bit from a word in a circulating dynamic storage loop, and giving a continuous output corresponding to this bit during each pulse time of one word period. A method for accomplishing this function is shown in Fig. 20. In this figure, Regenerative Memory Unit #1 is of the type illustrated in Figure 10 and previously described on page 19, and corresponds to the multiplier register in the block diagram of Fig. 19. Regenerative Memory Unit #2 is of the same type except that its loop delay is made as small as possible. This unit corresponds to the block labelled FF in Fig. 19. As in other systems herein described, synchronysm and phase coherence are maintained by the clock at f and oscillator at 2F. The "read-out" signal consists of a train of "zero" phase pulses with a single "one" phase pulse corresponding to the pulse time of the bit to be read out from regenerative memory unit 1. This read-out signal is fed to two places. First it is inverted and converted to pulse-no pulse script. The conversion is accomplished in a hybrid tee with an inverted clock input as shown. The output of the converter will then be a single pulse of "zero" phase during the time corresponding to the desired bit, and this will establish the phase of the CW oscillator. At all other times it will give no output and the CW oscillator will continue to oscillate with the established phase until it is changed by another pulse.

The read out signal is also sent to a phase script "AND" gate the output of which will be a train of "zero" phase pulses except during the time of the "one" phase pulse on the read out signal. During this time the output of the "AND" gate will be the same as the input from regenerative memory unit #1. The output of the "AND" gate is converted to pulse-no pulse script and after a short delay, is also used to establish the

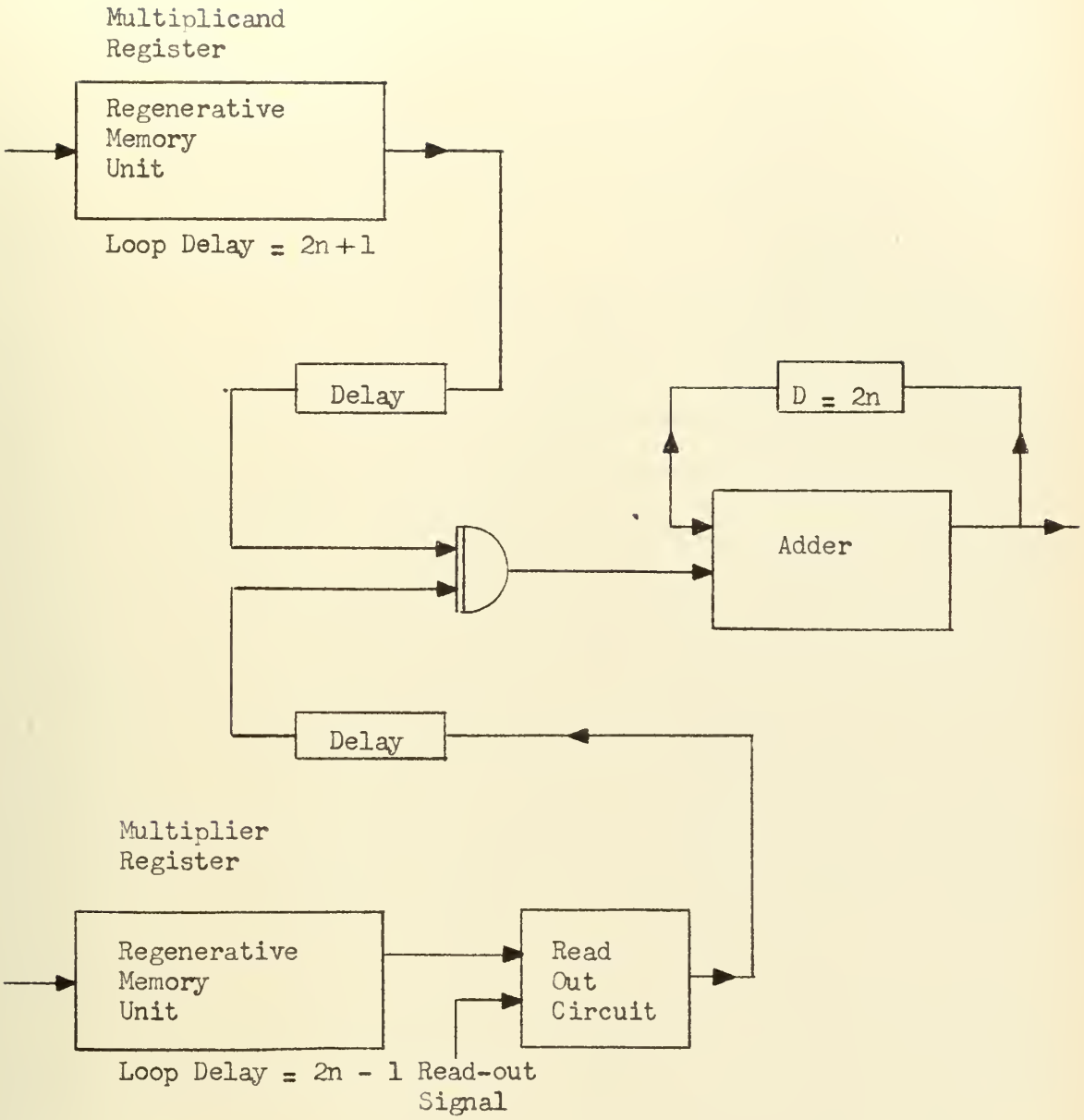


Fig. 21 - Block Diagram of a Microwave Phase Domain Multiplier

phase of the CW oscillator. Thus, the CW oscillator is in each case initially established in a "zero" phase oscillation. If the signal read out from regenerative unit #1 is a "one" phase pulse, the phase of the CW oscillator will be immediately changed to a "one" phase oscillation, while if the signal read out is a "zero" phase pulse, the phase of oscillation will remain unchanged.

The output of regenerative memory unit #2 will then be a continuous train of either "one" or "zero" phase pulses as determined by the phase of the CW oscillator, and hence will give repeated presentation of the bit taken from regenerative unit #1. It might be noted at this point that memory unit #2 acts as a single bit register even though the storage loop contains several bits since the CW oscillator maintains all pulses circulating in the loop of the same phase. Depending upon the development of a short delay traveling wave tube the loop delay of this register might be made small enough to make it in fact a single bit register. In this case, the CW oscillator would no longer be necessary since a single pulse would then be sufficient to establish the phase of the single circulating pulse.

We now have all the essential elements for a microwave multiplier, a block diagram for which is shown in Fig. 21. For the multiplier and multiplicand registers regenerative memory units with loop delays of 2n-1 and 2n +1 bits, respectively, are used. The read-out circuit for the multiplier digits is similar to that just described, and includes a "single bit" register in the form of a short delay regenerative memory unit. The adder could be similar to the one previously described in Fig. 13, with a feedback loop equal to 2n pulse times for recirculating the partial products.

If a word time is assumed equal to 2n pulse times (where n is the number of digits in the operands), then the read out signal will consist of a series of "zero" phase pulse except during the first pulse time of each word when a "one" phase pulse will occur. Thus, on the first pulse of each successive word time the next higher order digit of the multiplier will be read out from the multiplier register. The delays shown between the registers and the adder must be so adjusted that the first digit of the multiplicand arrives at the "AND" gate simultaneously with the first pulse from the single bit register in the read out circuit. At the end of n word times plus the delays involved in the multiplier circuit, the double length product will have been formed and will be circulating in the adder loop from whence it can be coupled out by suitable control signals.

3.7 Phase Script Arithmetic Unit

It is now possible to combine the units previously described into a complete arithmetic unit capable of performing algebraic addition, subtraction and multiplication. The logical design of such a unit can be found in Appendix III (see Fig. 40). Such an arithmetic unit might be expected to have an addition time of the order of 65 musec for algebraic addition of two 20 bit numbers. Multiplication in this unit of two 20 bit numbers would require of the order of 1625 musec.

The arithmetic unit shown in Fig. 40 would require about 90 magic-T's and 10 short delay, high gain, traveling-wave tube amplifiers. These figures, of course, account for only the circuitry shown and make no provision for such sophistications as error detection and correction or overflow control, nor do they include equipment required in selection networks and in generation of necessary control signals. Their significance

is therefore somewhat questionable. It is felt, however, that in conjunction with Fig. 40 they do serve to illustrate a principle by which microwave techniques can be used in algebraic manipulation of numbers and to indicate the order of magnitude of the speed which might be expected in performing arithmetic operations with microwave techniques. Here is a unit which is inherently capable of operating on 20 bit words at a rate of 15 million additions/second or 600,000 multiplications/second.

3.8 Other Phase Script Devices

There are of course, other devices necessary for the practical realization of a microwave computer, not the least of which is a pulse generator. Several references to millimicrosecond pulse generators can be found in current literature; however, to date none have achieved pulse widths of as short as 1 mμsec. An experimental regenerative pulse generator has been built by C.C. Cutler[4] which operates at 4000 mc and produces 3 mμsec pulses at a p.r.f. of about 14.5 mc. A pulse generator of the Cutler type for the generation of r.f. pulses of approximately 6 mμsec width at 9 kmc has been described by A.C. Beck[5]. A second type generator which generates pulses of equivalent size and frequency, but by a much simpler device, has been described by Beck and Mandeville[6]. In addition to the fact that they generate pulses which are not as narrow as we might like, the pulse generators mentioned above all have the characteristic that adjacent pulses may be of arbitrary r.f. phase, and therefore their use with a phase script would be prohibited. A pulse generator is currently under development at the General Electric Microwave Laboratory which it is hoped will overcome these difficulties. A diagrammatic representation of the proposed generator is shown in Fig. 22.

A CW oscillator operating at frequency F (say 10 kmc) is fed through a traveling wave tube amplifier which is gated by a CW signal at much

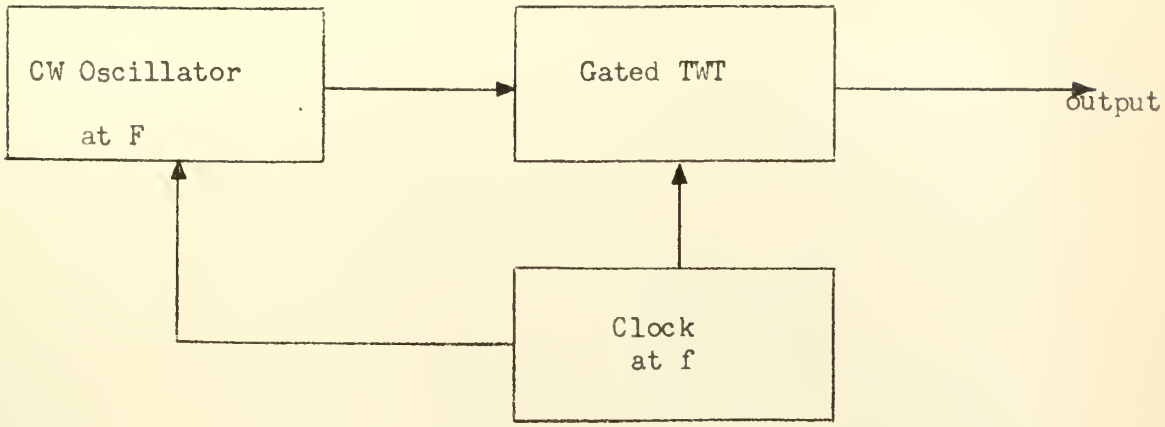


Figure - 22. Block Diagram of a Millimicrosecond Pulse Generator

lower frequency f (say 500 mc). The output will then be a train of 10 kmc pulses approximately 1 mμsec wide a repetition rate of 500 mc. The CW oscillator of frequency F is stabilized by the clock frequency f to insure that successive pulses at the output of the amplifier will be of the same r.f. phase. The gated traveling-wave tube amplifier will consist of a specially designed traveling wave tube with a grid cavity across which the clock signal at frequency f is impressed. The grid is biased so that oscillations at the clock frequency will cause the electron beam to be interrupted at this rate and hence cut off the gain of the tube.

In addition to a millimicrosecond pulse generator, one must have a means for reading information into the regenerative phase script memory unit previously described. A method by which information might be transferred from a static slower speed memory such as a magnetic core matrix to the high speed dynamic regenerative phase script memory is illustrated in Fig. 23. It is assumed that the information is initially stored in 20 K binary flip-flops which are controlled by a core memory where K is the number of words presented, and each word contains 20 bits. The clock

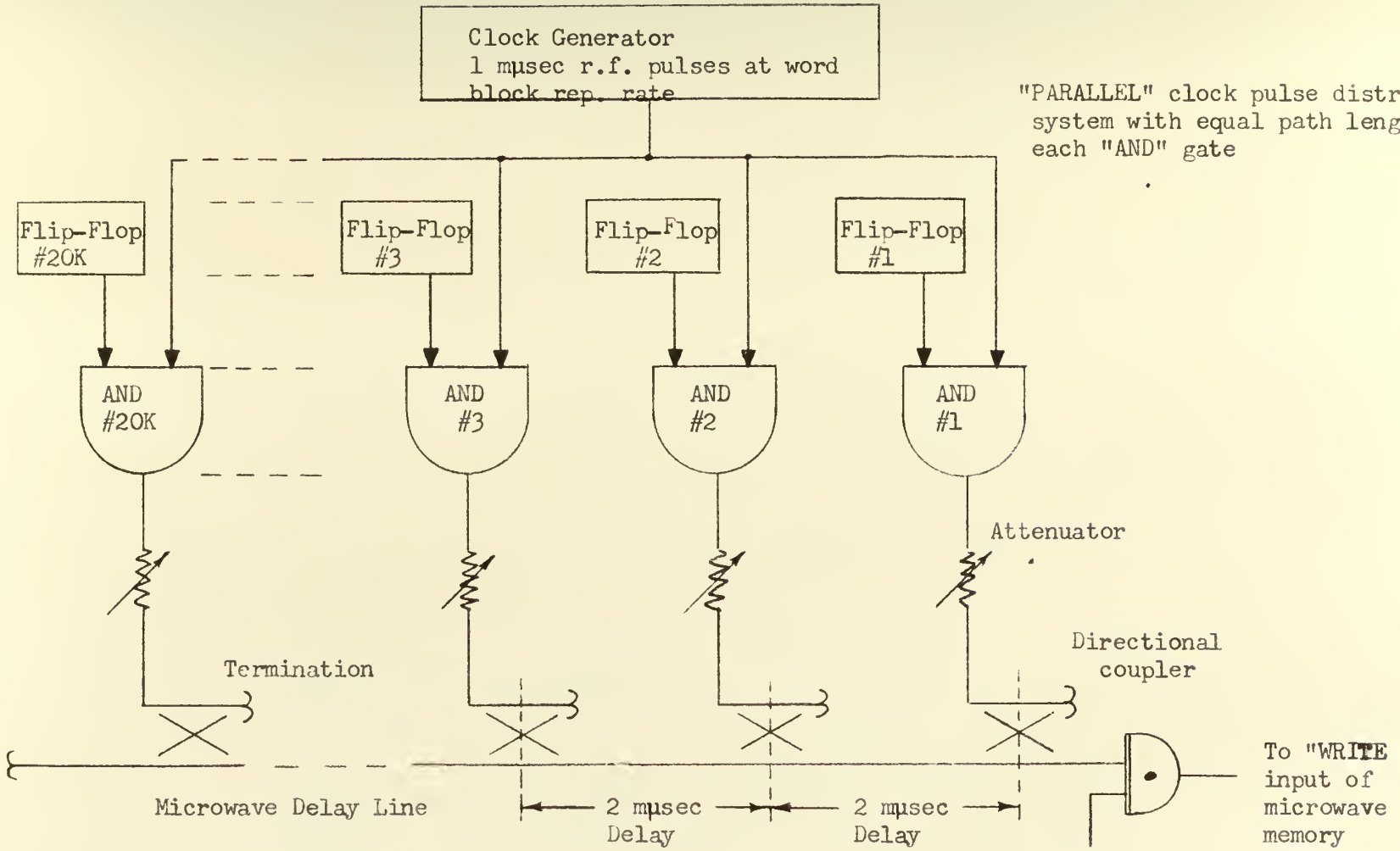


Fig. 23 - Circuit for Read-In to Microwave Memory from Low Speed Flip-Flops

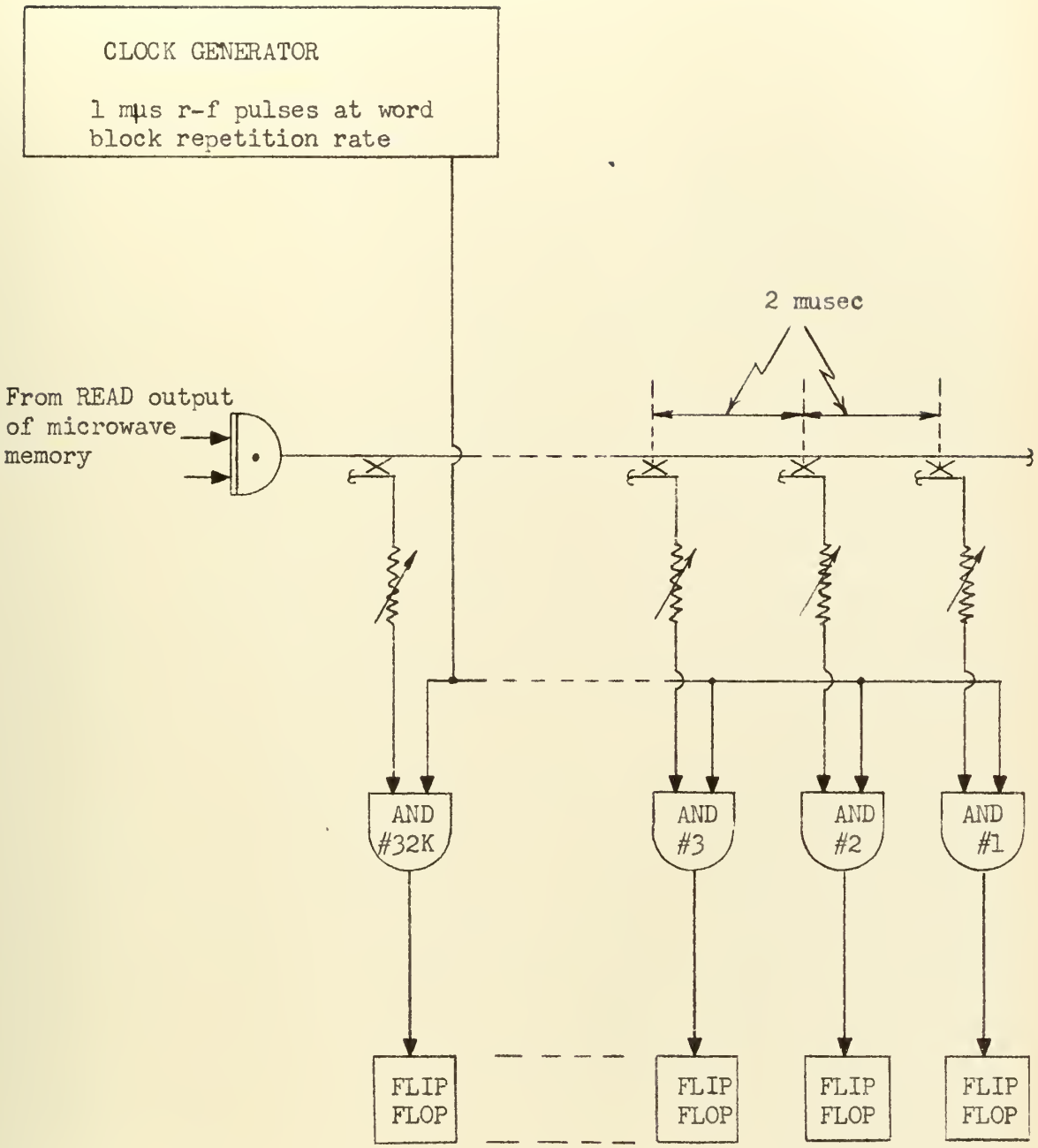


Fig. 24 - Circuit for Read-Out from Microwave Memory

generator provides 1 μsec r.f. pulses at word block repetition rate, i.e., at intervals of 2 x 20 x K μsec (if the bit period is 2 μsec).

The clock pulses are distributed to the 32K "AND" gates each of which contains one video and one r.f. input, so as to arrive at all gates simultaneously. Upon arrival of the r.f. pulse, each AND gate produces a pulse on the microwave delay line through a directional coupler, the phase of each pulse being controlled by the video input to its respective "AND" gate. Pulses on the delay line are then fed serially into a recirculating microwave memory unit through a phase script "AND" gate which is open for the duration of a full word block.

The 20 K "AND" gates having one video and one r.f. input can be realized by controlled (180^{o}) phase shifters which might be traveling wave tubes whose electrical length (phase shift) may be conveniently changed by varying helix voltage. Ferrite phase shifters have also been proposed for this use.

A method by which information might be read-out of a microwave phase script memory is illustrated in Fig. 24. Assuming that it is desired to read out a block of K 20 bit words, the phase script "AND" gate is opened for the duration of the full word block allowing the pulses representing the words to fill the delay line. Directional couplers spaced at 2 μsec intervals along the line feed the pulses to the 20 K "AND" gates. At the instant that the first pulse arrives at #1 "AND" gate the succeeding pulses are simultaneously arriving at their respective "AND" gates. A 1 μsec pulse from the clock generator applied at this instant simultaneously to all "AND" gates will allow all pulses to be passed to their respective flip-flops. The flip-flops in this case would be CW oscillators, locked to twice the pulse carrier frequency so that they have two stable phase states, as was previously described in connection with the regenerative memory unit.

CHAPTER IV

FREQUENCY DOMAIN TECHNIQUES

No discussion of microwave computer techniques would be complete without mention of those techniques which make use of the frequency domain wherein a frequency script is used to represent the information in the r.f. pulses. That is, the information content of the r.f. pulses is contained in the frequency of oscillation of the electrical current of each pulse, rather than in its amplitude or phase. Extensive analyses of frequency domain devices in the low frequency range have been carried out in recent years, [10] - [17] primarily at the Stanford University Electronics Laboratories led by Dr. W.A. Edson. Much of the application for frequency domain devices is to be found in the electronic countermeasures field; however, the invention of the frequency-memory register by Dr. Edson at Stanford University introduced the interesting possibility of assembling frequency domain devices into computing circuitry. The use of the Frequency domain in electronic digital computers has been extensively covered in a report by K. Amo.[10] This report includes, in addition to theoretical analysis and descriptions of many frequency domain computer devices reports on experimental work using such devices. Amo's work was done in the region of low frequency, but the implications of possible transposal to the microwave region are clearly indicated.

The term "frequency-memory" has generally been given to a class of multi-mode oscillator circuits which have the two following rather remarkable properties:

(a) The circuit is capable of continuous oscillation at any one of several different frequencies, and it will oscillate at only one of these mode-frequencies at a time.

(b) The circuit may be forced into continuous oscillations at any particular mode by the injection of a signal pulse at the mode frequency.

One of the attractive features offered by such frequency memories from the computer standpoint is the possibility of conveniently operating in a decimal radix rather than a binary, for if a frequency memory can be made with ten stable mode frequencies it can be used as a decimal register.

The basic idea of frequency memory at low frequency can be conviently illustrated by the circuit shown in Figure 25. If it is assumed here that the antiresonant circuits are of comparable selectivity and

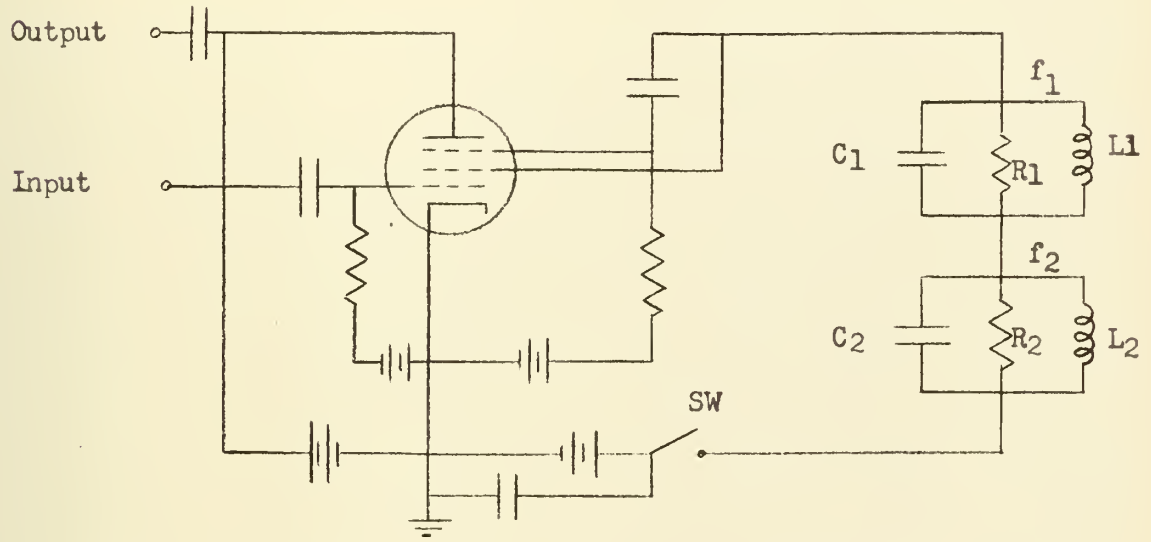


Fig. 25 - Two frequency oscillator illustrating Frequency Memory concept

impedance and are tuned to frequencies that are unrelated but of the same order of magnitude, then oscillation at either f_1 or f_2 can be initiated by supplying to the input a signal of suitable magnitude and the desired frequency. An input of short duration suffices, for once started the oscillation persists without change until the other frequency is injected or the power is turned off. In this manner, it can be said the circuit remembers the frequency of the last input.

At microwave frequencies a frequency memory might more conveniently take the form shown in Figure 26. In this circuit the frequencies at which stable oscillations may occur are determined by the total loop delay.

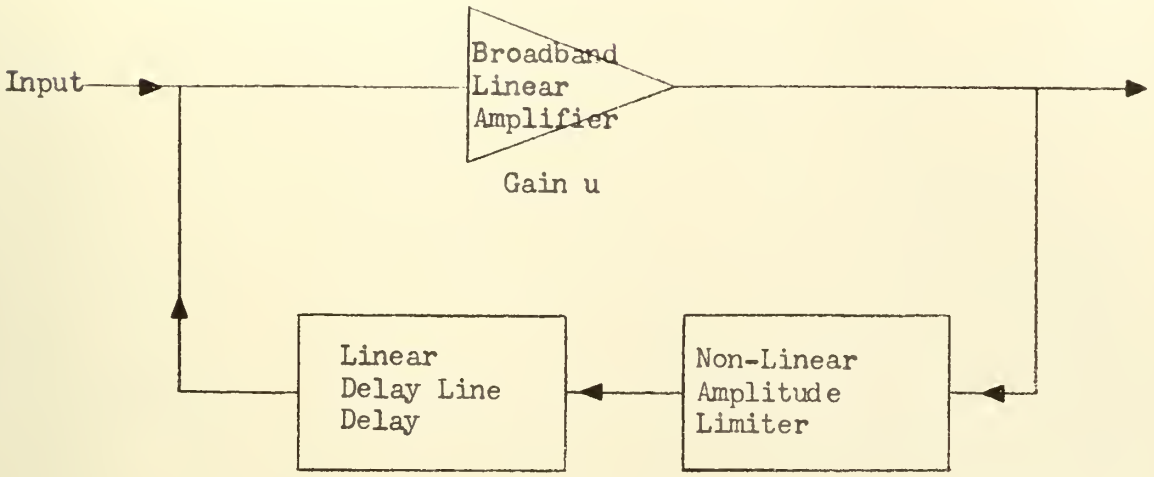


Fig. 26 - Block Diagram of Microwave Frequency Memory

A signal introduced into the input circulates around the loop in the indicated sense and stable modes of oscillation may exist at each frequency for which, first, the loop gain, μ, is larger than unity, and, second, the loop phase shift is a multiple of 2π radians.

Theoretical analyses of multimode oscillator circuits and development of stability criteria are extensively covered in the literature and will not be presented here. It will be informative, however, to take a closer look at the factors which determine mode spacing and read-out and instruction times in multimode oscillators since a general understanding of these will be helpful in promoting an understanding of means in which these devices are used in computing circuits.

Instruction signals will normally consist of a short pulse of the frequency to be stored, f_1, as illustrated in Figure 27(a). Such a pulse has a power spectrum as shown in Figure 27(b).

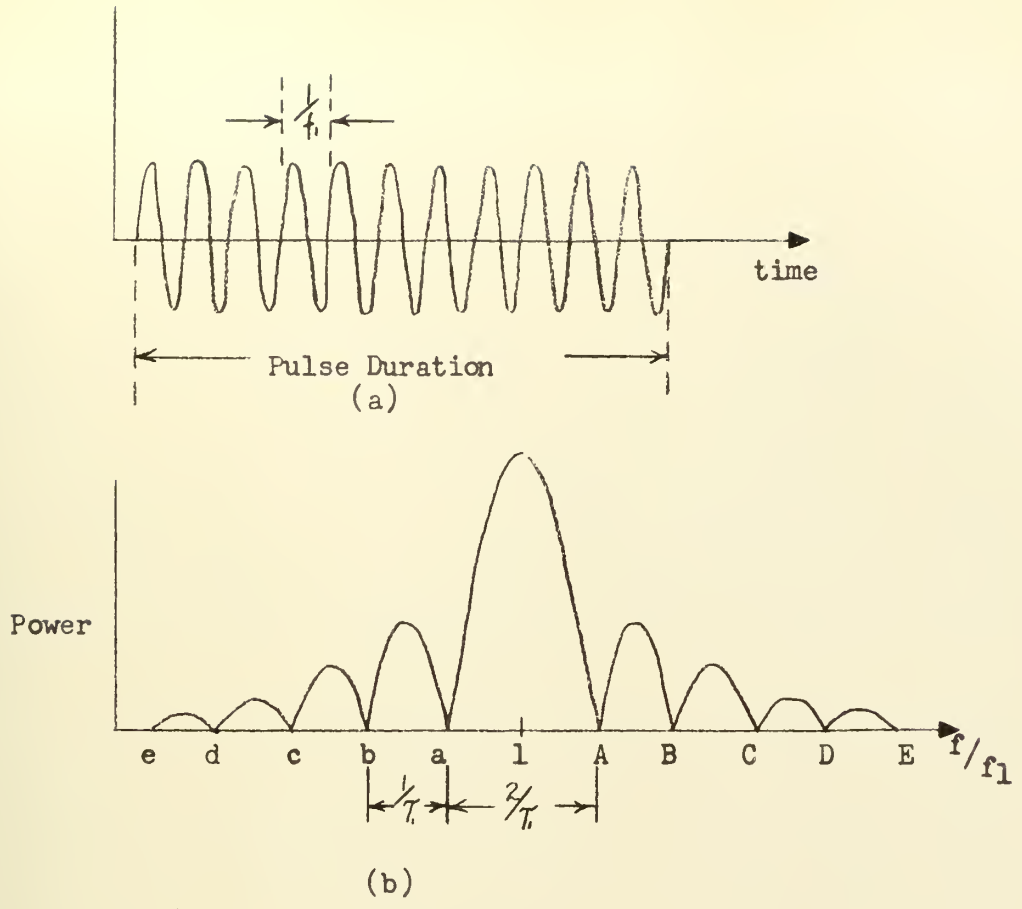


Fig. 27 (a) - Frequency Memory Instruction Pulse

(b) - Power Spectrum of Pulse in (a)

Since the power in the instruction pulse is spread over a continuous frequency spectrum as shown in Figure 27(b), the mode frequencies of the frequency memory must be far enough apart so that more than one mode will not be excited by the same instruction pulse. For the ideal case, the mode frequencies different from f_1 should correspond to the frequencies labelled A, a, B, b, etc., in Figure 27(b), since there is no power in the instruction pulse at these frequencies and maximum power at f_1. Hence, for this case, we see that the minimum mode spacing of a frequency memory is determined by the duration of the instruction pulses to which it will be required to respond, thus $\Delta f = \frac{1}{\tau_i}$, where Δf is the minimum mode spacing.

The above discussion was concerned with the problem of matching the instruction time and the mode spacing in order to insure exciting of only one mode. There is also the question of the time or power required to excite the mode to a sufficient degree so that sustained oscillations occur. This question is difficult to solve analytically because non-linear properties of the memory circuit must be considered. A plausible estimate for the instruction time of a memory system at rest seems to be that the instruction time be equal to the frequency memory delay time. In the case of switching from an existing oscillation into a new mode, Amo [10] shows that the instruction time should be about twice the loop delay time.

In consideration of the read-out time of a frequency memory the Fourier power spectrum is again helpful. If the frequency memory oscillations are observed for a time τ_1 , a pulse will be observed with a power spectrum similar to that of Figure 27(b) where the null power frequencies are spaced $\frac{1}{\tau_1}$ cycles apart. In order to determine which of the mode frequencies is in oscillation, τ_1 , must be long enough that $\frac{1}{\tau_1}$ covers a bandwidth including only one mode. Thus the minimum read-out time is related to mode spacing by

$$\tau_{1_{min}} = \frac{1}{\text{mode spacing}}$$

Frequency domain devices based on the frequency memory have been discussed extensively in the literature [9] - [17] and numerous methods have been proposed for using these devices to perform arithmetic operations. Several of these proposals will be reviewed here to illustrate how frequency domain devices will be used at microwave frequencies in digital computers. For illustrative convenience, a frequency memory device consisting of a multi-mode S-band oscillator is assumed to have ten possible

mode frequencies from 2.7 kmc through 3.6 kmc spaced 100 megacycles apart, representing the digits 0 through 9 as follows:

Mode Freq (kmc)	2.7	2.8	2.9	3.0	3.1	3.2	3.3	3.4	.35	3.6
Digit assigned	0	1	2	3	4	5	6	7	8	9

Further, it is assumed as discussed above that the minimum instruction time of this one-decimal-digit storage device is 10 μsec, and hence, pulse width will be assumed to be 10 μsec.

4.1 The Counting Operation

A counting circuit consists basically of a frequency memory which must have the following three capabilities:

(1) switch from operating mode to next higher one upon arrival of a trigger signal

(2) switch from 9 to 0 by the same trigger signal

(3) when switching from 9 to 0, produce a trigger signal to act upon another decade counter

A possible microwave counter circuit proposed by M.P. Forrer[9] is shown in Figure 28. For illustrative purposes assume that the frequency memory is initially oscillating at a frequency of 3.5 kmc, representing number 8. This frequency passes through the frequency shifter and is shifted to 3.6 kmc and is then available at the input to gate "A". An instruction pulse of 7.4 kmc is applied to gate "A" which opens the gate allowing transmission of the 3.6 kmc signal to the balanced modulator where it is combined with a 1 kmc signal. Of the three output frequencies from the balanced modulator, 2.6, 3.6, and 4.6 kmc, only the signal at 3.6 kmc is passed by the band-pass filter. This signal then forces the frequency memory into oscillation at 3.6 kmc, representing number 9, and hence one pulse has been counted.

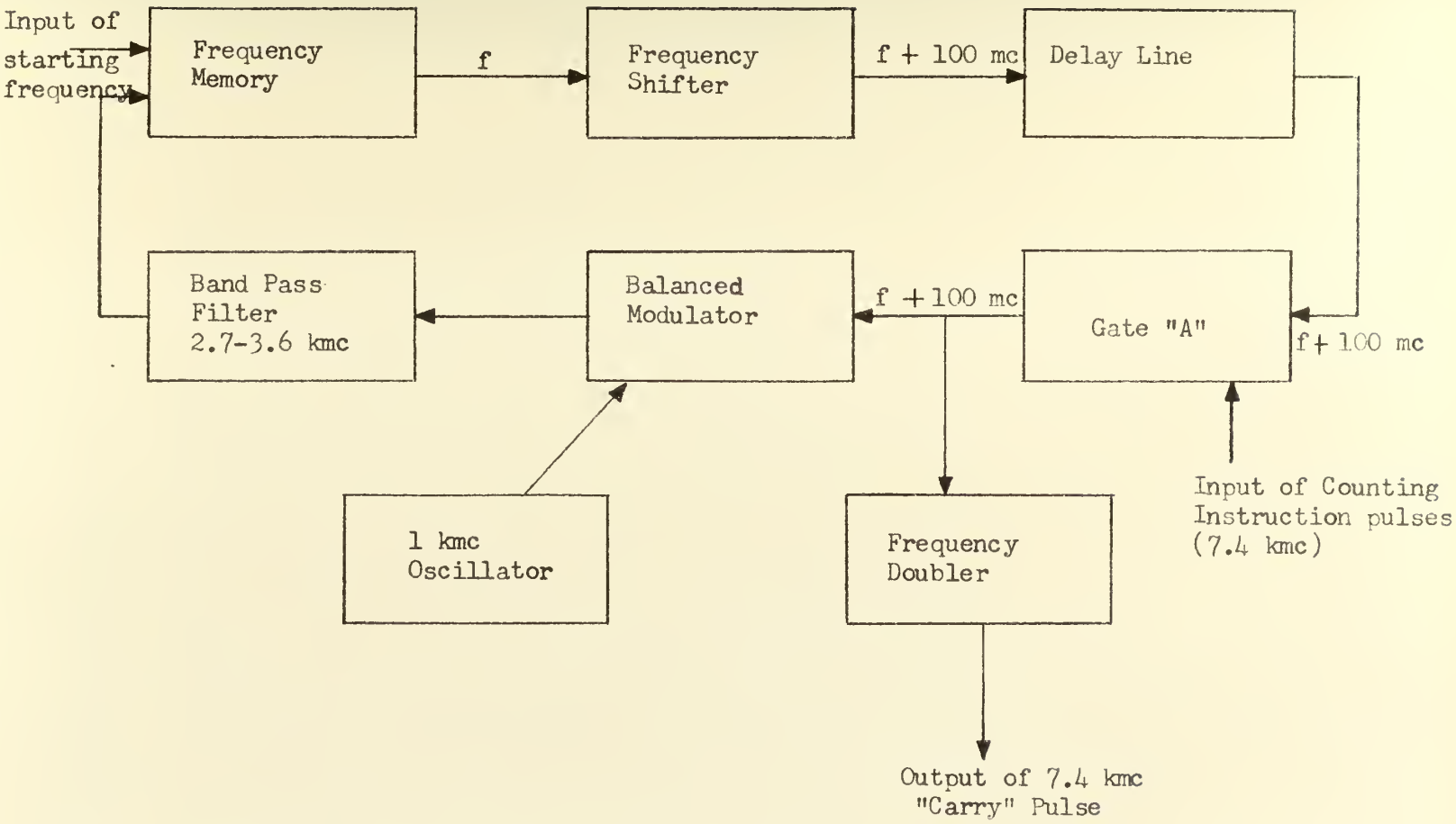


Fig. 28 - Block Diagram of a Frequency Domain Counting Circuit

If another instruction pulse at 7.4 kmc is now applied to the counter a slightly modified reaction occurs which causes the frequency memory to shift from 9 to 0 and also provide a single pulse at 7.4 kmc as an instruction pulse for a similar counter in the next higher order. In this case the signal appearing at the input to gate "A" is 3.7 kmc. At the output of gate "A" a portion of the 3.7 kmc signal is doubled and becomes a 7.4 kmc "carry" pulse which is used as an instruction pulse for the next counter. It should be noted that although "carry" pulses occur at other times, they are of different frequencies, and it is only pulses at 7.4 kmc which are capable of opening gate "A" and hence acting as instruction pulses. When the 3.7 kmc from gate "A" is combined in the balanced modulator with 1 kmc the resulting signals are 2.7, 3.7, 4.7 kmc, and only 2.7 kmc is passed by the band pass filter and hence the frequency memory is forced to oscillate at 2.7 kmc, representing number 0.

A circuit for realization of gate "A" is illustrated in Figure 29. This circuit has the characteristic that the signal on one input line will appear at the output only if a 7.4 kmc signal appears on the other input line. The two inputs, say 3.5 and 7.4 kmc, are combined in the balanced modulator. Of the outputs from the balanced modulator, only the 10.9 kmc signal will be passed by the high pass filter. This is combined with 7.4 kmc in the second balanced modulator. Of the output frequencies only 3.5 kmc will be passed by the low-pass filter. When the instruction signal is other than 7.4 kmc, no output will occur because of the action of the filter circuits.

The frequency shifter mentioned above in discussing the counter circuit can be realized in numerous ways. One possibility is illustrated in Figure 30. In this circuit the input signal f is combined in a balanced modulator with an arbitrary frequency f_a. The high pass filter passes

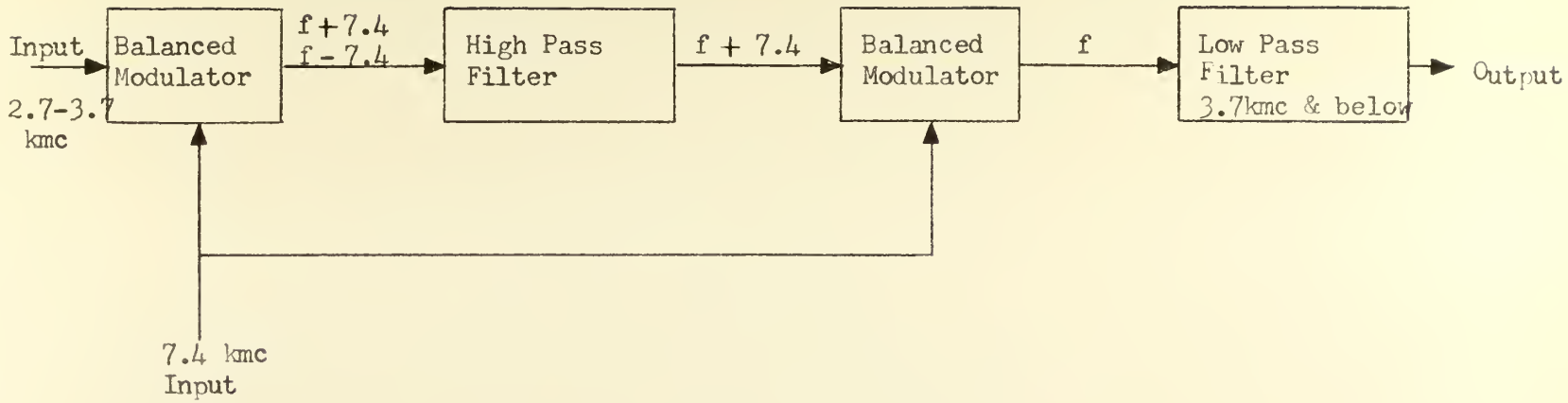


Fig. 29 - Circuit for Realization of Gate "A" of Figure 44

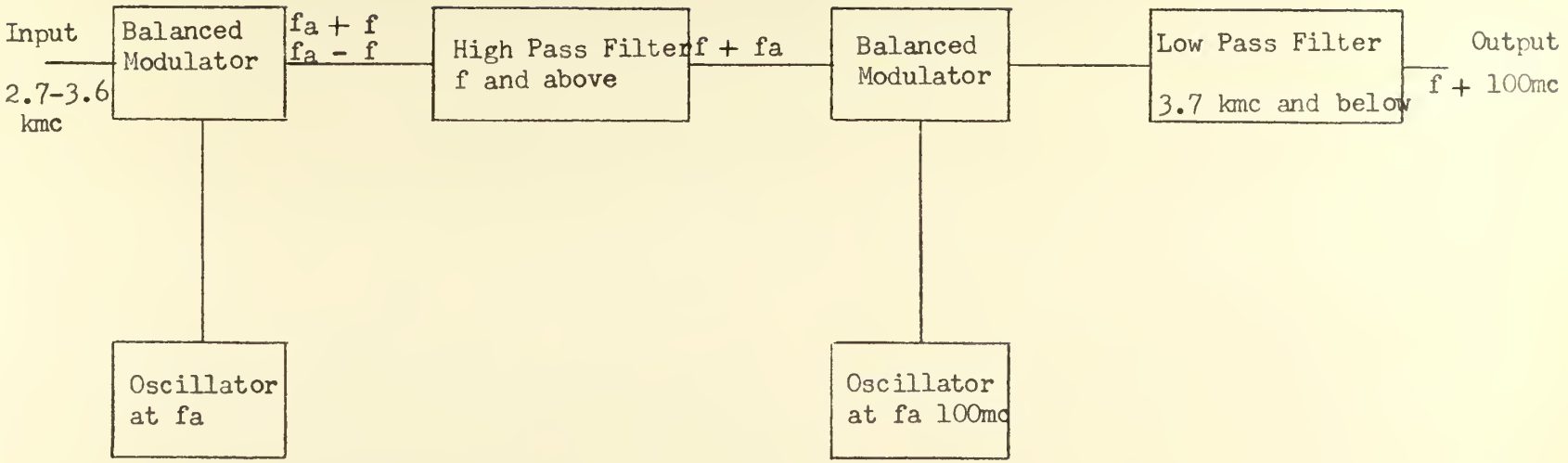


Fig. 30 - Block Diagram of a Frequency Shifter

fa + f which is then combined with fa - 100 mc. The low pass filter passes (f + fa) - (fa - 100) = f + 100 and the desired result is achieved.

The maximum operating speed of the frequency domain counter shown in Figure 28 can be assumed to be determined by the total loop delay. Reasonable estimates of the delay of the various elements might be as follows:

Frequency Memory	10 μsec
Frequency Shifter	5 μsec
Other circuit elements	5 μsec

These estimates give a total loop delay of 20 μsec, and hence the maximum counting speed might be expected to be of the order of 50 million counts per second.

3.5 Frequency Domain Adder

The process of addition in the frequency domain is basically a problem of adding frequencies. This can be conveniently done by means of balanced crystal modulators. Addition can be carried out in either a parallel or series fashion. The speed of a parallel frequency domain adder is primarily determined by the carry propagation time through the various orders of the adder. Since this carry propagation is of necessity basically a serial process, the speed of a parallel frequency domain adder is not significantly greater than that of a serial adder. Furthermore, the serial adder provides a very large saving in the amount of equipment required. For these reasons, frequency domain addition will be illustrated with a serial adder.

Realization of a serial adder requires the use of a different type of frequency memory than was previously described. This new form of the frequency memory is referred to as a multifrequency memory. This device

is merely an extension of the previously discussed frequency memory and consists of a closed loop containing a delay line, a gate, an amplifier and a frequency memory all in series as illustrated in Figure 31. If a pulse of alternating current with the desired shape and having one of the permitted frequencies (as determined by the frequency memory) is introduced at the input it will be propagated around the loop until power is removed or until it is overridden by another input pulse occurring at the time the circulating pulse reenters the amplifer. Other input pulses may be stored in the loop on a time division basis. As each pulse circulates around the loop it will be amplified and reshaped during each recirculation. The number of pulses which may be stored in this device is determined by the time delay of the loop and by the space necessary between pulses.

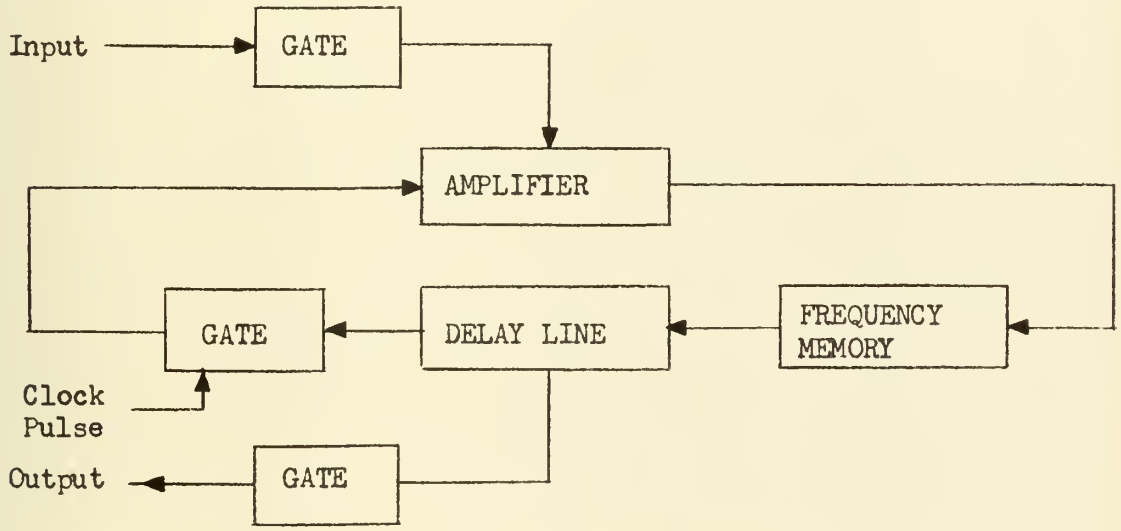


Fig. 31 - Block Diagram of a Multi-Frequency Memory

The loop is essentially a serial storage device in that a given pulse is only available at the output once during each recirculation. It should be noted that due to the properties of the frequency memory the circulating pulses are reshaped at each recirculation. Dispersion in

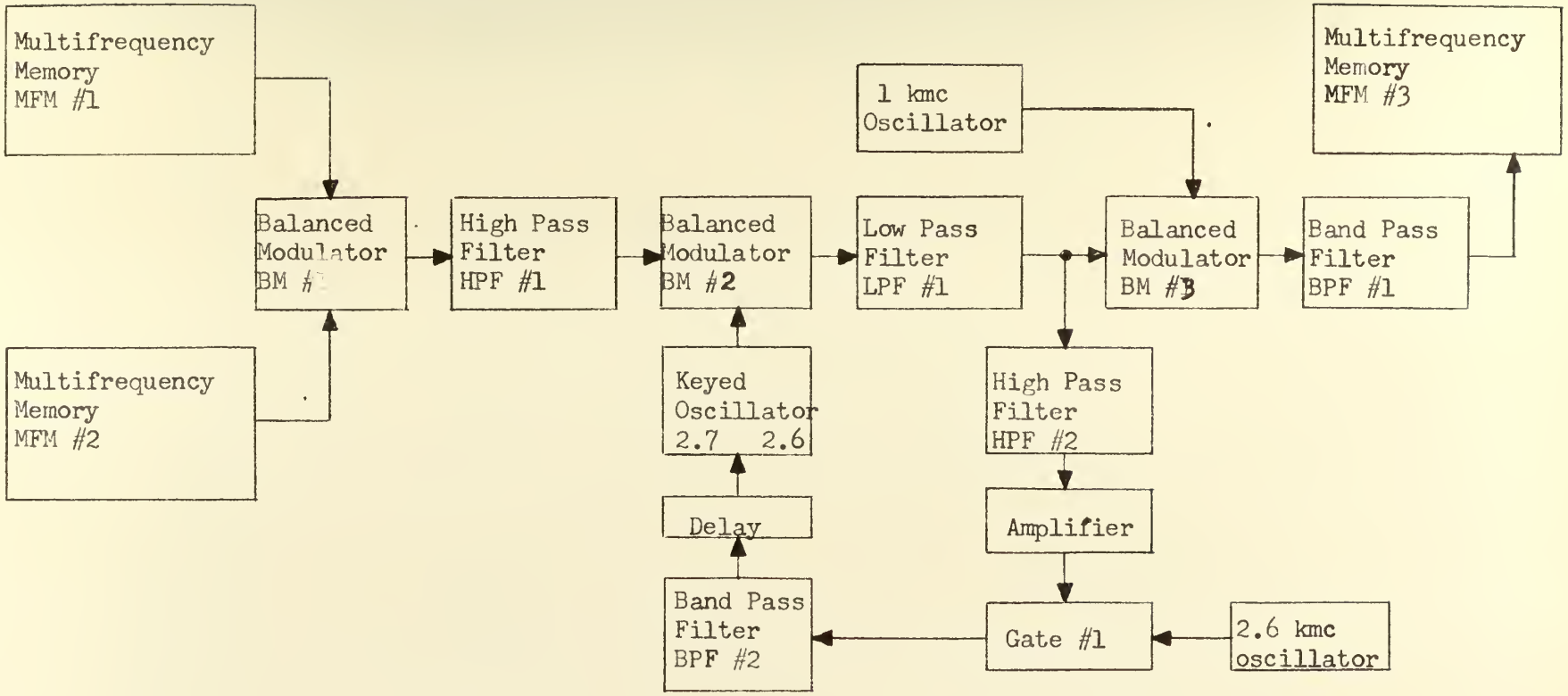


Figure 32 - Block Diagram of Frequency Domain Serial Adder

the delay line thus has no cumulative effect upon the waveforms. For the example to follow it will be assumed that the multifrequency memory is capable of storing more than ten separate pulses (digits) in the recirculating loop.

A block diagram for a serial adder is shown in Figure 32. The three multifrequency memories are of the type described in Figure 31 and are assumed to be operating in exact synchronism. The keyed oscillator oscillates at 2.7 kmc except when synchronized by a 2.6 kmc signal, during which time it oscillates at 2.6 kmc. The pass band of the various filters shown in Figure 32 are given below.

HPF #1 - Pass 5.4 kmc and above
Stop 3.6 kmc and below

HPF #2 - Pass 3.7 kmc and above
Stop 3.6 kmc and below

LPF #1 - Pass 4.6 kmc and below
Stop 5.4 kmc and below

BPF #1 - Pass 2.7 kmc to 3.6 kmc

BPF #2 - Pass 2.6 kmc

Pulses representing the two numbers to be added are stored in multifrequency memories #1 and #2. Assume for illustrative purposes the following example:

$$\begin{array}{r} 295 \\ \underline{373} \\ 668 \end{array}$$

During the first digit period pulses representing the lowest order digits (3.4 and 3.2 kmc) are mixed in Balanced Modulator #1. The sum of these two frequencies (6.2 kmc) then appears at the output of HPF #1. In BM #2 6.2 kmc mixes with 2.7 kmc from the keyed oscillator. The difference

frequency 3.5 kmc is passed by LPF#1 and goes through BM#3 and BPF#1 to MFM#3. Thus, a pulse at 3.5 kmc representing number 8, the lowest order digit of the sum, is stored in MFM#3. No signal is passed by HPF#2.

During the second digit period pulses of 3.6 kmc and 3.4 kmc representing 9 and 7 and presented to BM #1. The sum, 7.0 kmc, appears at BM #2 and is mixed with 2.7 kmc. The difference frequency, 4.3 kmc, appears at the input of BM #3 where it is mixed with 1 kmc. The difference, 3.3 kmc (number 6), passes BPF #1 to MFM #3 and is stored as the second digit of the sum. The 4.3 kmc output from LPF #1 is in this case passed by HPF #2 and amplified. This amplified signal opens the gate which allows a 2.6 kmc pulse to switch the keyed oscillator to 2.6 kmc. If the delay, D, is properly adjusted so that the 2.6 kmc input to BM #2 coincides with the pulse representing the next order digit coming from HPF #1, the substitution of 2.6 kmc for 2.7 kmc in BM #2 corresponds to adding an additional 1, thus introducing a carry from the addition of the two lower order digits (in this case 7 and 9). The addition of the following digits occurs in like manner and a series of pulses representing the sum of the two input numbers will be available in MFM #3.

The addition time for a serial frequency domain adder such as that described above is, of course, dependent upon the number of digits in the operands and the time spacing between digits. The minimum allowable time between digits is determined by the delay in the "carry" circuit in that a "carry" pulse must have sufficient time to switch the keyed oscillator to 2.7 kmc before the next higher order digit pulse enters BM #2. If we assume a 20 musec delay around the carry loop, this gives a corresponding digit period of 20 musec. Assuming further a seven digit decimal number we find the addition time to be of the order of 7 x 20 = 140 musec.

The techniques for utilizing frequency domain devices in digital computers have been illustrated in performing two operations, counting and addition. It should now be apparent that by judicious combinations of such devices as frequency memories, balanced modulators, directional couplers, fixed frequency oscillators and r.f. pulse controlled gates, circuits could be designed to perform other arithmetic operations as well. Numerous illustrations of means for accomplishing multiplication, division and subtraction have been given by K. Amo [10].

It has been stated that one of the attractive advantages of the frequency domain technique is the ability to operate with a decimal radix rather than a binary radix. Aside from the obvious convenience of operating with a decimal radix, an increase in computing speed is apparently available when serial operation is employed because of the smaller number of digits required to represent a given number in a decimal radix. For example, a 7 digit decimal number could require as many as 23 binary digits to represent it, and hence, assuming the computing elements used in each case operated at comparable speeds, the addition of two numbers in decimal form could be accomplished in about 1/3 the time required for the addition of the same two numbers in binary form. On the other hand, there is a not too obvious disadvantage of using a decimal radix. That is, that for a given available bandwidth the information flow rate capability of frequency domain devices is greater with a binary radix than with a decimal radix. Consider, for example, that a bandwidth of 2000 mc is available in the microwave region. For a decimal radix, the maximum frequency separation between digits is then 200 mc if all ten possible frequencies are to be contained within the 2000 mc bandwidth. A rectangular pulse of width τ_1 will have a power spectrum as illustrated in

Figure 43(b). In order to determine the frequency of such a pulse as when reading it out from a frequency memory, the pulse width must be of sufficient length that the bandwidth $\frac{1}{\tau_1}$ as illustrated in Figure 43(b) is narrow enough that a sensing circuit can identify its center frequency as one of the permitted frequencies. If the pulse is too short $\frac{1}{\tau_1}$ will be large enough that appreciable power is present at two or more of the mode frequencies and the sensing circuit will not be able to tell which one it is. Hence, it appears that the maximum allowable bandwidth, $\frac{1}{\tau_1}$ of any pulse would be 200 mc. If we assume a pulse with a power spectrum such that there is no power in the pulse at 200 mc intervals above and below the center frequency we find that the required pulse length is $\frac{1}{200 \times 10^6} = 5$ μsec. Allowing equal time for spacing between pulses gives a pulse period of 10 μsec.

On the other hand, if the same bandwidth of 2000 mc is assumed with storage in binary form, the maximum allowable frequency separation between digits is now 1000 mc. By the same process as in the preceding example it can be shown that the pulse bandwidth may now be as large as 1000 mc and hence the pulse length may be reduced to $\frac{1}{1000 \times 10^6} = 1$ μsec. Once again allowing equal time for spacing between pulses gives a pulse period of 2 μsec. In comparing the storage capacity in the examples given above it will be noted that in a given time, say 20 μsec, only 2 decimal digits could be presented. This compares to 10 binary digits which could represent a 4 digit decimal number as high as 2047. Thus it can be seen that because of the increased flow rate possible with a binary radix, frequency domain devices might actually be expected to compute faster with a binary rather than a decimal radix.

CHAPTER V

SUMMARY AND CONCLUSIONS

In the preceding chapters numerous illustrations have been given to indicate that microwave devices, when combined to form computing circuits, are inherently capable of performing logic and other computer functions at rates considerably in excess of those now employed or envisioned in the immediate future. Of the three basic methods presented by which information may be contained within r.f. pulses none seems to exhibit such superiority as to warrant its selection as the most likely to succeed. Frequency domain techniques have been more extensively investigated than the other two. Such techniques, however, would require the use of considerably more equipment in the form of fixed frequency oscillators, balanced modulators, filters and the like, and appear to lack the flexibility in performing logic which is available when the more conventional AND/OR gate type operations are used. The ability to operate with a decimal radix offers no particular advantage in that an expected speed advantage from the fact that fewer digits are required to represent a given number is offset by the longer time required for sensing the frequency of a given pulse. Phase script techniques are currently attractive because of the relative simplicity of the gating elements required and the ease with which negation may be accomplished. Whether such accurate control of the phase of r.f. pulse can be conveniently and practically accomplished, however, is a problem to be considered. Pulse-no pulse techniques, although inadequately represented in this paper cannot be lightly dismissed. The development of gating elements which respond purely to a pulse-no pulse script would remove the requirement for precise phase control encountered with the phase script.

Practical realization of any microwave computer techniques is not a prospect for the immediate future. Many years of development will be required on such devices as millimicrosecond pulse generators, short delay, high gain traveling wave tubes, broad band microwave limiters and delay lines. Although the cost will be high, it must be paid if such ultra-high computing speeds are to be attained.

It is interesting to attempt a rough estimate as to what would be required in the way of presently available components, such as high-speed transistors, to match the inherent speed of a microwave computer in performing arithmetic operations. Such equivalent speed, of course, could be obtained only by carrying the concept of parallel operation to ridiculous extremes.

As an example, one might consider logic similar to that proposed by Weinberger and Smith[18] for high speed addition in which a one microsecond adder using one megacycle circuitry is proposed. In this instance, rapid addition is accomplished through simultaneous carry generation. If we consider the use of high speed transistor circuitry using 5 mcs transistors to implement the logic described, one might expect to accomplish addition in about 0.2 μsec. For a 21 bit word, the circuitry described would require in the neighborhood of 800 transistors. In order to match the inherent 15 million additions/sec of the microwave unit, three such transistor adder units would be required, a total of about 2400 transistors. At an estimated cost of $10.00 each this would require $24,000 for transistors alone.

From the standpoint of multiplication one might consider the "simultaneous" multiplier illustrated by C.K. Richards[19], who describes this

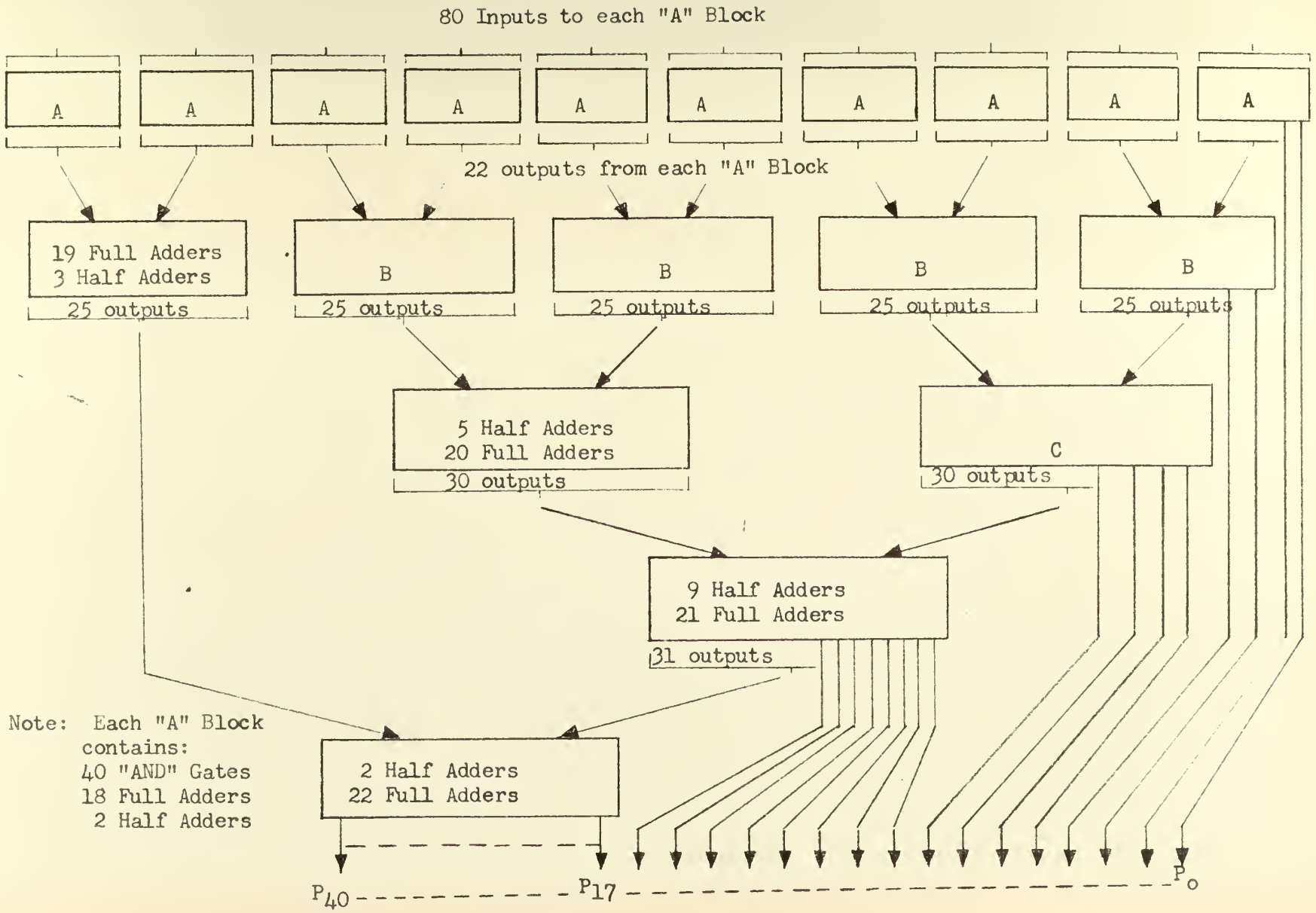


Fig. 33 - Block Diagram of a Simultaneous Multiplier Circuit.

circuit as the fastest multiplier known. A block diagram for such a multiplier which provides for the simultaneous multiplication of two 20 bit words is shown in Fig. 33. Such a configuration would require in the neighborhood of 9,000 transistors, and would be capable of producing the double length product in approximately 3 μsec. If the principles of simultaneous carry generation discussed above were incorporated into the simultaneous multiplier circuit a reduction in multiply time to a little more than 1 μsec could be attained. Such a modification could be made by increasing the total number of transistors to about 16,000. Thus the speed of the microwave multiplier could be equalled with transistors costing of the order of $80,000.

The cost of obtaining high operating speeds through parallel operation is thus seen to be high. In all fairness, however, it must be mentioned that at the present $1,000 per tube cost for traveling-wave tube amplifiers, even a simple microwave unit would not be inexpensive. Traveling-wave tubes are, however, still in a comparatively early stage of development. Given the demand which successful development of microwave computer techniques could well provide, there is no reason to believe that their cost could not be substantially reduced.

Ultimate utilization of the inherent operating speeds of microwave arithmetic unit requires that means be provided for transferring information to and from such a unit at comparable speeds. Experience has shown that arithmetic unit operating speeds are not the primary factor in determining the overall work rate of a computer system. Instead, the ability to obtain data and instructions from the memory is frequently the speed-limiting factor. The storage of information in the form in which it would be used in a microwave arithmetic unit can be accomplished

in a transmission line. The length of line required per bit storage and the bulkiness of the line itself, however, make any large capacity storage in such form highly undesirable. Utilization of high capacity storage devices such as magnetic cores, therefore, requires the use of conversion devices that will convert information stored in these conventional devices to the form in which it will be used in the microwave unit. One form which such a device might take has been previously discussed and is illustrated in Figure 23. Such a device, however, would still be incapable of sustaining sufficiently high information flow rates from the cores to the microwave unit to completely utilize the inherent operating speed of the microwave unit. It would seem, therefore, that lacking a complete microwave system, a microwave arithmetic unit could be most efficiently used as an ultra-high-speed unit in a large computing system. Such a microwave unit would have a small buffer microwave memory of the delay line type of sufficient capacity to support the microwave arithmetic unit during the cycle time of the conventional memory.

BIBLIOGRAPHY

1. M.P. Forrer, Patent Disclosure Letter on a Microwave Inhibit-Gate, No. 36-66D-41, December 1957.

2. M.P. Forrer, Patent Disclosure Letter on "AND" and "OR" Gates for Microwave Phase Domain Computer, No. 36-66D-17, September 16, 1957.

3. W.A. Edson, Patent Disclosure Letter on a Regenerative Memory Unit for Phase Domain Computer, PD-72, January 15, 1957.

4. C.C. Cutler, The Regenerative Pulse Generator, Proc. IRE, Vol. 43, February 1955.

5. A.C. Beck, Microwave Testing with Millimicrosecond Pulses, Trans. IRE, Vol MTT-2, April 1954.

6. A.C. Beck & G.D. Mandeville, Microwave Traveling-Wave Tube Millimicrosecond Pulse Generators, Trans. IRE, Vol. MTT-3, December 1955.

7. M.P. Forrer, Microwave Computer Research, Quarterly Progress Report through 31 July 1957, General Electric Microwave Laboratory Report No. R 57EL79, 23 August 1957.

8. M.P. Forrer, Microwave Computer Research Quarterly Progress Report through 31 October 1957, General Electric Microwave Laboratory Report No. R 57ELM79-1, 5 December 1957.

9. M.P. Forrer, Investigation of Application of Frequency Memory Techniques, General Electric Microwave Laboratory Report No. R55 ELS 12.3, 12 December 1955.

10. K. Amo, Use of the Frequency Domain in Electronic Digital Computers, TR No 470-1, Stanford University, December 31, 1956.

11. M. Disman, Registers and Counters Based on Frequency Memory, TR-No.19, Stanford University, 16 August 1954.

12. W.A. Edson, Frequency Memory in Multi-mode Oscillators, TR-No.16, Stanford University, July 19, 1954.

13. R.W. DeGrasse, Stability of Multi-Mode Oscillatory Systems, TR-No.18, Stanford University, August 9, 1954.

14. H.C. Lee, Linear Analysis of Multi-Mode Oscillatory Systems, TR-No.20, Stanford University, July 26, 1954.

15. H.C. Lee, A Flip-Flop Circuit Based on Frequency Memory, TR-No.81, Stanford University, January 20, 1955.

16. L.D. Smithey, The Traveling-Wave Amplifier as a Multimode Oscillator, Thesis, U.S. Naval Postgraduate School, Monterey, California, 1956.

17. V. Met, On Multimode Oscillators with Constant Time Delay, Proc. IRE, August 1957.

18. A. Weinberger and J.L. Smith, A One-Microsecond Adder Using One-Megacycle Circuitry, IRE Trans. Vol EC-5 June 1956.

19. R.K. Richards, Arithmetic Operations in Digital Computers, D. Van Nostrand Co., New York, 1955.

20. E.C. Nelson, An Algebraic Theory for use in Digital Computer Design, Trans IRE, Vol. EC-3, September 1954.

21. R. Serrell, Elements of Boolean Algebra for the Study of Information-Handling Systems, Proc. IRE, October 1953.

22. J.R. Pierce, Traveling-Wave Tubes, D. Van Nostrand Co., 1950.

23. G.G. Bower, The Traveling-Wave Tube as a Computer Component, NAVORD Report 4565, U.S. Naval Ordnance Laboratory, Corona, 24 January 1956.

24. S. Sensiper, Electromagnetic Wave Propagation on Helical Structures (A Review and Survey of Recent Progress), Proc. IRE, February 1955.

APPENDIX I

THE USE OF BOOLEAN ALGEBRA IN COMPUTER LOGICAL DESIGN

Boolean Algebra can be a most useful tool in the design of the logic circuitry of a digital computer operating in a binary mode. Any computer process can be described by a set of statements which specify its logical properties, i.e., the result, or output variable, of the process is described in terms of the various logical combinations of the input variables which combine to produce the given result. Thus, the statements describing the computer process are translated into a set of algebraic equations. Since the output and input variables of a binary computer process can assume only two possible values (0 and 1), these algebraic equations can be manipulated into various forms by the use of Boolean Algebra. Finally, the algebraic operations in the equations can be interpreted in terms of specific computer elements and thus lead to the realization of a circuit for the original process. It should be understood from the start, however, that such procedures will not necessarily lead to the "best" circuit for a particular process. What the algebra does provide is a convenient means of representing a switching circuit without drawing the circuit. Also, and probably more important, is the fact it provides a means for quickly finding a multitude of different circuits that will perform any desired switching function. With a little practice, the circuit designer thereby has a powerful tool to aid him in finding a "good" circuit, even though it may not be the best one.

Basic Principles of Boolean Algebra

In Boolean Algebra, the variables can have only two discrete values, 0 and 1, and as in ordinary algebra, symbols may be used to represent the

variables. There are two basic operations, called addition and multiplication. Addition is represented by a plus sign ($+$) and has the meaning of "OR". Thus, the symbolic equation $A + B = C$ has the meaning that $C = 1$ if either A OR B is 1; otherwise $C = 0$. Multiplication is represented by a (x) or dot($\cdot$) and has the meaning of "AND". The equation $C = A \cdot B$ thus has the meaning that $C = 1$ only if A AND B both are 1; otherwise $C = 0$. A third operation which is found in Boolean Algebra, and which has no counterpart in ordinary algebra is the operation of complementation, commonly and conveniently designated by a bar over the symbol ($\bar{A}$) and having the meaning of "Not A". It can easily be shown by use of the above definitions that the Boolean "AND" and "OR" operations are commutative, associative, and distributive. The following list of relationship, all of which can be obtained from the above definitions, will further illustrate the properties of Boolean Algebra.

$$A + 0 = A$$

$$A + 1 = 1$$

$$A + A = A$$

$$A \cdot 0 = 0$$

$$A \cdot 1 = A$$

$$A \cdot A = A$$

$$A + (B \cdot C) = (A + B)(A + C)$$

$$\overline{A \cdot B} = \bar{A} + \bar{B}$$

$$\overline{A + B} = \bar{A} \cdot \bar{B}$$

$$A + (A \cdot B) = A$$

$$A(A + B) = A$$

$$A + (\bar{A} \cdot B) = A + B$$

$$A(\bar{A} + B) = A \cdot B$$

$$(A + B)(\bar{A} + C)(B + C) = (A + B)(\bar{A} + C)$$

$$(A \cdot C) + (\bar{A} \cdot B) + (B \cdot C) = (A \cdot C) + (\bar{A} \cdot B)$$

$$\overline{A \cdot B \cdot C} = \bar{A} + \bar{B} + \bar{C}$$

$$\overline{A + B + C} = \bar{A} \cdot \bar{B} \cdot \bar{C}$$

Another interesting property of Boolean functions which is often quite useful in determining alternate forms for a given function is that referred to as "Duality". This property is described as follows: If in an algebraic expression, each addition is replaced by a multiplication, each multiplication is replaced by an addition, and each signal variable is replaced by its complement, the resulting expression is the complement of the original expression; e.g.,

$$\overline{A \cdot B} = \bar{A} + \bar{B}$$

$$\overline{A + B} = \bar{A} \cdot \bar{B}$$

$$\overline{A \cdot (B + C)} = \bar{A} + (\bar{B} \cdot \bar{C})$$

Applications to Computer Design

In the application of Boolean Algebra to computer design, it is first necessary to establish circuits which realize the "AND", "OR" and "NOT" functions previously described. With these circuits then a direct correlation can be obtained between the algebraic equations and the actual computer circuits. The actual physical form which these circuits take is, of course, dependent upon the manner in which signal variables are actually represented electrically in the computer circuit (i.e., voltage, current, phase, etc.). However, the circuits can be functionally represented, and as such are referred to as "gates". Hence, an "AND" gate is a circuit such that a signal representing 1 will appear at the output terminal only if a signal representing a 1 is applied to all of the input terminals.

An "OR" gate would represent a circuit which would produce an output signal representing a 1 in response to application of such a signal to any of the input terminals. A "NOT" gate, or "INVERTER" represents a circuit which gives a "1" output in response to an "0" input.

In the process of deriving an algebraic expression to represent a given computer process, it is convenient to make use of a correspondence table which shows the relationship between the outputs and inputs of the process. It should be apparent that with n inputs to a given process there are 2^n different input conditions which might exist. Hence, for a complete description of the process, an output condition must be designated for each of the 2^n possible input conditions. Each of the input combinations may be represented by an "AND" term such as $(\overline{A} \cdot B \cdot \overline{C})$ which has the meaning that a signal is applied to input B, but not to inputs A or C. An equation representing the entire process may then be written by listing all the combinations of input signals which will produce an output signal. Since the listing implies an "OR" relationship, it follows that the process may be represented by an expression of the form,

$$\ldots (\overline{A} \cdot \overline{B} \cdot \overline{C}) + (\overline{A} \cdot B \cdot \overline{C}) + (A \cdot \overline{B} \cdot C) \ldots$$

where only those terms which are to yield an output signal are included.

As an example of this procedure consider the process described by the correspondence table shown in Figure 34. This table lists the nine possible input combinations of three variables, A, B, and C and the output, X, corresponding to each of these input combinations. Thus, the Boolean expression describing this system could be written

$$X = (\overline{A} \cdot \overline{B} \cdot C) + (A \cdot B \cdot \overline{C}) + (A \cdot B \cdot C)$$

which means that X is 1 for any of the three input combinations listed, i.e., A and B are 0 and C is 1, OR A and B are 1 and C is 0, OR A and B and C

INPUT CONDITIONS			OUTPUT CONDITIONS
A	B	C	X
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Fig. 34 - A Correspondence Table

are all 1. This expression, of course, is not in its simplest form, but by making use of the properties listed previously it can be readily reduced to $X = (A \cdot B) + (\bar{A} \cdot \bar{B} \cdot C)$ and such a circuit could be realized by the combination of "AND" and "OR" gates as shown in Figure 35.

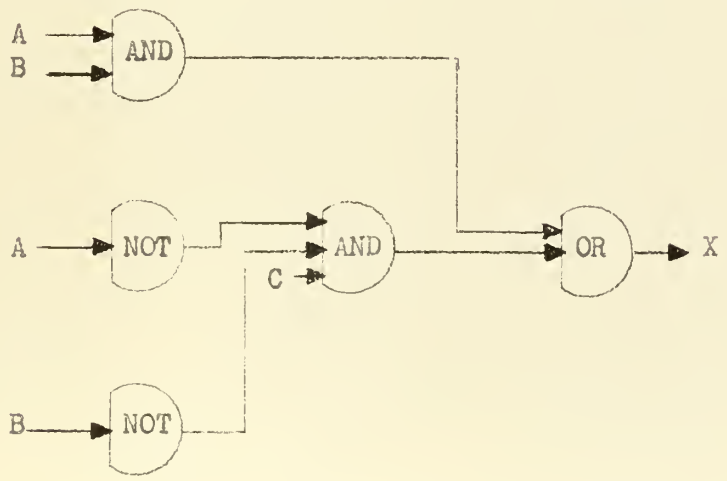


Fig. 35 - Logical Circuit for Realization of the Boolean Expression $X = (A \cdot B) + (\bar{A} \cdot \bar{B} \cdot C)$

APPENDIX II

The Traveling-Wave Tube as a Computer Component

1. Simple Description of Theory of Operation.

The adaptability of the traveling-wave tube as a computer component stems from its basic capability for producing high gain over wide frequency ranges without requiring the change of any mechanical tuning mechanism. Power amplification greater than 40 db over a 2:1 frequency range has been obtained.

The traveling-wave tube commonly takes the form of a helical transmission line arranged concentrically with an electron beam. Radio frequency energy essentially travels at the velocity of light along the wire from which the helix is wound. Since the wire and the r.f. energy follow this helical path, the actual progress of the energy along the axis of the helix is at some fraction of the velocity of light. This velocity is determined by the helix dimensions (i.e., its circumference and pitch) and by the dielectric loading due to the structure which supports the helix. The fields associated with this "slow-wave" extend inward into the center of the helix and there interact with the electron beam.

If electrons are sent along the axis of the helix at essentially the same velocity as the waves, an interaction between the waves and electrons occurs. This interaction results in a transfer of energy wherein the r.f. wave on the beam and helix grows at the expense of the d.c. beam energy. The electrons on the average are slowed down and give up just enough energy by this slowing down process to account for the increasing energy in the r.f. waves on the helix.

In practice, the electron beam is formed in a gun and is focused down the center of the helix to a collector electrode on the far end by the confining forces of a longitudinal magnetic field of a few hundred

gauss. The velocity of the electrons is determined by the voltage difference between the cathode in the electron gun and the helix, and this is adjusted to give the electrons just the right velocity for interaction with the waves. The signal to be amplified is coupled onto the end of the helix nearest the electron gun and propagates along the helix in the same direction as the electron beam. Because of the interaction, the fields on the helix grow exponentially with distance and these amplified waves are coupled off of the helix at the end farthest from the electron gun.

The devices used to couple the r.f. energy to and from the helix are special directional couplers which are in themselves helices. These helices, which are outside of the vacuum envelope of the tube are approximately matched to the input and output coaxial transmission lines so that fairly uniform coupling may be achieved over a broad band of frequencies comparable to the amplification band of the traveling-wave tube.

A theoretical discussion of traveling-wave tube operation can be broken into two parts, (a) the theory of the interaction of the electron beam with the electromagnetic wave, and (b) the theory of the propagation of the electromagnetic wave along the guiding structure. The combined effects of these two interactions must be included in any theoretical analysis of the traveling-wave tube. The usual procedure is to consider each of the problems separately and to combine them by superposition to get the overall result.

2. Interaction of Electron Beam and Electromagnetic Wave.

A simplified theory of the interaction of an electron beam with an electromagnetic wave on a generalized slow-wave transmission network is given by Pierce [22]. This theory is the small signal theory which means that the equations governing electron flow have been linearized by neglecting certain quantities which become negligible when signals are small.

This development shows that the interaction of the electron beam and the initial electromagnetic wave results in the formation of three waves which are propagated along the slow wave structure, each wave having an initial amplitude equal to 1/3 the amplitude of the original wave. The first wave is an increasing wave which travels a little more slowly than the electrons. The second wave is a decreasing wave which travels more slowly than the electrons. The third wave is an unattenuated wave which travels faster than the electrons. As these waves travel toward the output the first wave, growing in amplitude, will ultimately predominate, and the other components will become of vanishingly relative size. Under this condition, the gain of the tube will be that of the increasing wave. A general relation for this gain G expressed in decibels is $G = A + BCN$ decibels where A is a loss relating the initial voltage of the increasing wave to the total applied voltage, B is a figure describing the rate of growth of the wave, N is the length of the tube in wavelengths, and C is the gain parameter which is determined by the characteristics of the "slow-wave" structure and by the d.c. beam current and voltage. The gain parameter C will be more fully discussed below. If it is assumed that the initial electron velocity is equal to the velocity with which the waves are propagated in the tube in the absence of the electron beam and that the slow-wave structure of the tube is lossless, then A is typically about -9.54db and $B = 47.3$, and for this case $G = -9.54 + 47.3CN$ decibels.

In the use of the traveling wave tube as a computer component we are acutely interested in the time delay through the amplifier tube. It would therefore seem desirable to have an expression which indicates the relationship between gain and time delay. If t_d is the delay through the tube and f is the operating frequency, then the gain formula above may be

written

$$G = -9.54 + 47.3C f t_d$$

The gain parameter C is shown by Pierce [22] to be equal to the quantity $\left(\frac{KI_o}{4V_o}\right)^{1/3}$, where I_o and V_o are the beam current and accelerating voltage, and K is the helix impedance, a factor which is determined by the characteristics of the helix. Proper evaluation of this helix impedance requires investigation of the propagation of an electromagnetic wave along a helix.

3. Propagation along a Helix

The problem of propagation of an electromagnetic wave along a helix has been approached by numerous methods and has been reported in a large number of papers in recent years [24]. Although an exact solution has not been obtained, Pierce [22] shows that to a satisfactory approximation the solution may be obtained by considering a helically conducting cylindrical sheet. The sheet is perfectly conducting in a helical direction making an angle ψ, the pitch angle, with a plane normal to the axis (the direction of propagation) and is non-conducting in a helical direction normal to this direction. The results of such an analysis are expressed in terms of three phase or propagation constants. These are

$$\beta_o = \frac{\omega}{c} \qquad \beta = \frac{\omega}{v} \qquad \gamma = \sqrt{\beta^2 - \beta_o^2} = \sqrt{1 + \left(\frac{v}{c}\right)^2}$$

Here c is the velocity of light and v is the phase velocity of the wave. β_o is the phase constant of a wave traveling the speed of light. β is the actual axial phase constant and γ is the radial propagation constant. Actually, for phase velocities usually used (as determined by accelerating voltage V_o) γ and β are essentially equal.

In his analysis Pierce shows that various field components vary as modified Bessel functions of the argument γr, where r is the radius, and hence results appear in the form of curves of various functions plotted against γa, where a is the mean helix radius.

Figure 36 gives information concerning the phase velocity of the wave. The coordinates are generalized to apply to any helix, however the abscissa is proportional to frequency and the ordinate is essentially the ratio of the wave velocity to the velocity the wave would have if it traveled along the helically conducting sheet with the speed of light in the direction of conduction. From this curve it can be seen that there is a region where the velocity of the waves varies as a function of frequency labelled the "dispersive region", and there is a region where the velocity of the waves is essentially independent of frequency. It is in this region, known as the "non-dispersive region", where a broadband amplifier is normally operated. The significance of this curve may perhaps be made more clear by reference to Figure 37. This figure shows the same curve as Figure 36 plotted for specific helix dimensions ($a = 0.75$ mm; $\cot \psi = 16$). The abscissa is now labelled directly in frequency while the ordinate gives the voltage required to match electron velocity with wave velocity.

The bandwidth of a traveling-wave tube is in part determined by the range over which the electrons keep in step with the wave. These curves then indicate why a helix type traveling wave tube can amplify over wide bandwidths without changing helix voltage. In the "non-dispersive region shown in Figure 36 a constant helix voltage maintains the wave and electron velocities in synchronism over a broad frequency range.

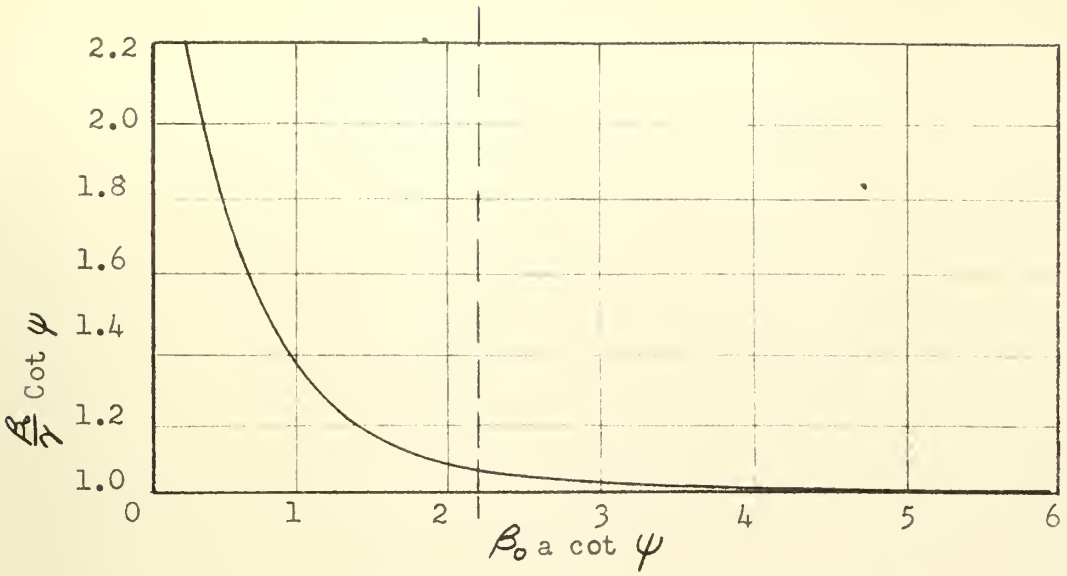


Fig. 36 - Traveling Wave Tube Dispersion Curve

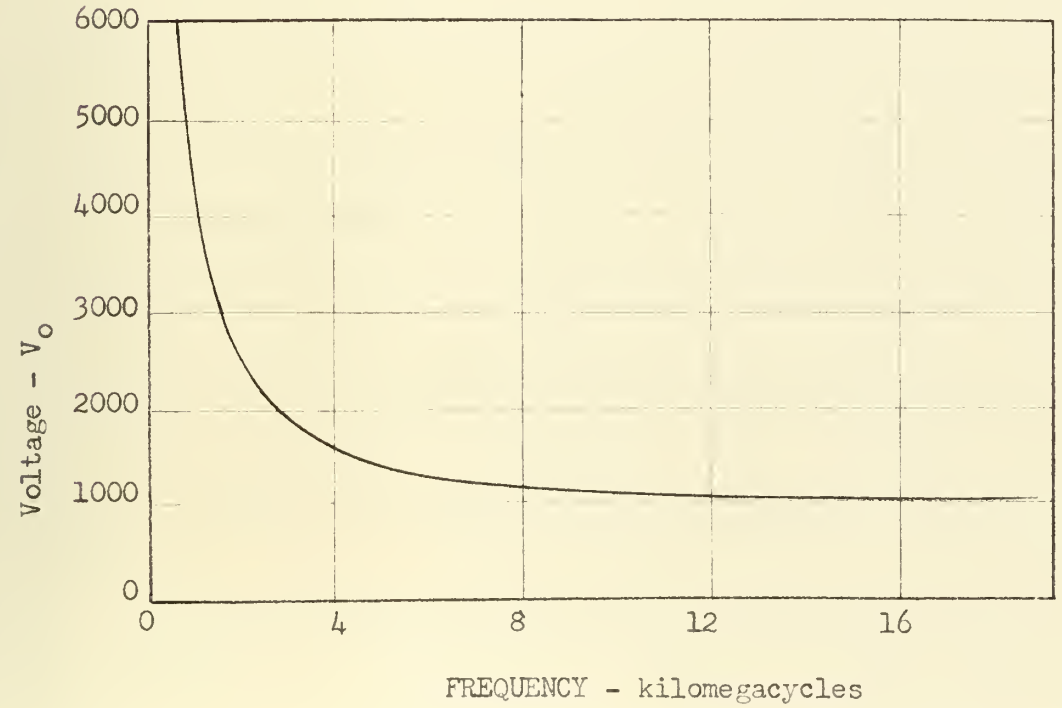


Fig. 37 - Variation of Synchronous Voltage with Frequency for a Typical X-Band Traveling Wave Tube

Figure 38 gives information concerning the helix impedance in the form of a plot of $K\beta_o/\gamma$ as a function of γa for various values of the ratio of electron beam radius to mean helix radius. For any given values of γa and b/a, the helix impedance K can be determined by multiplying the ordinate by γ/β_o.

It is interesting to note from Figure 36 that in the non-dispersive region (and this is the only region in which we are interested for computer applications of the traveling-wave tube) the ordinate $\beta_o/\gamma \cot\psi$ is essentially equal to 1. Hence, $\beta_o \cot\psi \approx \gamma$, and the abscissa is essentially γa. This then indicates a minimum value for γa for broadband operation. In addition, since $\cot\psi \approx \gamma/\beta_o$, the helix impedance is obtained by multiplying the ordinate from Figure 38 by $\cot\psi$.

From the gain equation $G = -9.54 + 47.3Cft_d$ it can be seen that if high gain is desired with short delay time (t_d) for a given frequency, the gain parameter C must be made as large as possible. This in turn requires a high value of helix impedance. Figure 38 shows that high values of K are obtained by making b/a as close to 1.0 as possible and having a small value of γa. It was shown above, however, that for broadband operation, the dispersion curve in Figure 36 indicated a minimum value for γa. This, therefore limits K,C and hence the gain. It should be noted, also that since $C = \left(\frac{KI_o}{4V_o}\right)^{1/3}$ high beam currents and low beam voltages also help to improve gain.

It is interesting and informative to compute the gain for a given set of conditions. Assume that at a frequency of 10 Kmc it is desired to determine the maximum gain which might be obtained with a 1 μsec delay time. Further assume Vo = 1000 volts and Io = 10 ma. If we assume synchronism between the electron velocity and wave velocity, then the wave

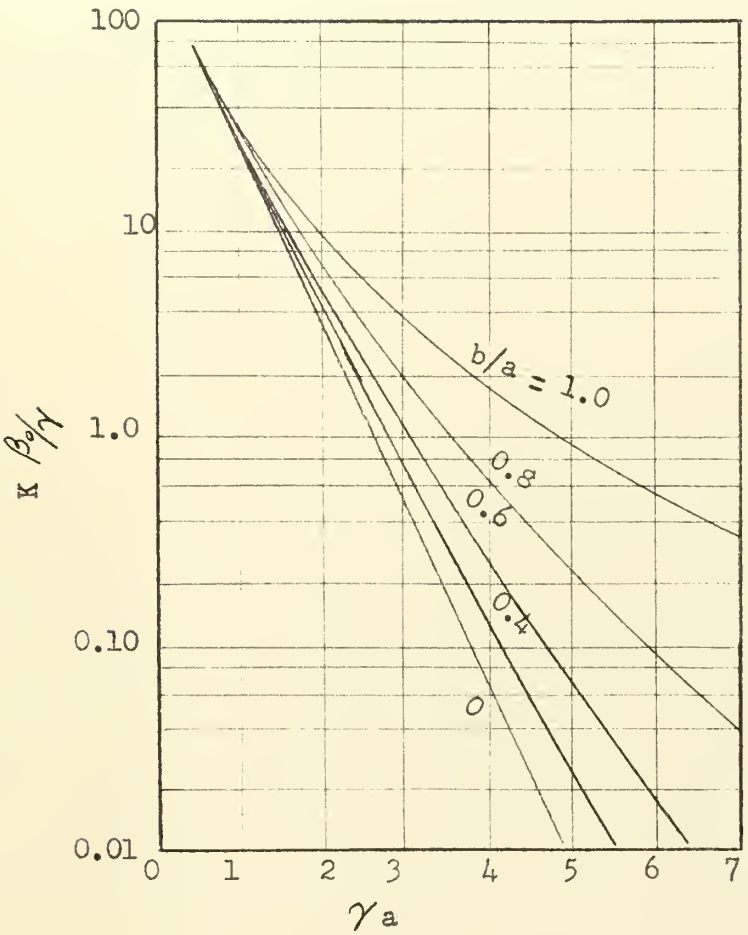


Fig. 38 - Circuit Impedance K for a solid beam of electrons of radius a and propagation constant γ under the condition that electron velocity is equal to the velocity of the undisturbed wave

velocity and hence γ are essentially determined by Vo. Furthermore, $\cot\psi$ is approximately equal to c/v and hence it too is determined by Vo. With Vo = 1000, it turns out that $\cot\psi = 16$ and $\gamma = 3340$. From the dispersion curve of Figure 36 we take the minimum γa for broadband operation to be about 2.5. From Figure 38, and assuming b/a = 0.8 we get $K\frac{\beta_e}{\beta} = 2.8$ and hence $K = 3.1 \cot\psi = 49.5$

the gain parameter is

$$C = \frac{KI_o}{4V_o}^{1/3} = \frac{(49.5)(10^{-2})}{(4)(10^3)}^{1/3} = .050$$

and the gain is

$$G = -9.54 + (47.3)(.05)(10^{10})(10^{-9})$$

$$G = 14 \text{ db}$$

Although the preceding calculations indicate that a gain of 14 db, or even higher is theoretically possible with such extremely short time delays as 1 μsec, the attainment of such gain in practice presents numerous problems. The values of γ and γa shown above indicate a helix radius, a, of .75 mm. With Io = 10 ma and b/a = 0.8, a beam current density of 0.7 amp/cm^2 is indicated, a figure which is perhaps pushing the limits of practicality. Furthermore, with such high current density and such a small helix, very strong magnetic fields and considerable precision adjustment would probably be required to maintain the electron beam within the helix. It must also be noted that no losses were considered in these calculations.

Presently available commercial traveling wave tubes do not approach these theoretical values of gain and short delay. It is interesting and informative to compute theoretical values for gain and time delay for a commercially available tube using the actual tube characteristics in the computations and to compare these computed values with values actually

obtained. A typical X-Band commercial tube has the following characteristics

Io = 2.5 ma	$\cot\psi$ = 14.6
Vo = 1200 volts	a = 0.8 mm
Effective Helix Length = 190 mm	b/a = 0.8
Actual Helix Length = 216 mm	γa = 2.5
N = 92.5	v = 2.05×10^7 m/sec
Gain = 40 db	f = 10 Kmc

From Figure 38 we can determine K and thereby compute C

$$K = (3.1)(14.6) = 45.4$$

$$C = \left[\frac{(45.4)(2.5)\,10^{-3}}{(4)(1200)}\right]^{1/3} = .0286$$

The theoretical gain then is

$$G = -9.54 + (47.3)(.0286)(92.5) = 125 \text{ db}$$

This compares with a measured gain at 10 kmc of 40 db. From the point of view of time delay we can solve the gain equation for t_d

$$t_d = \frac{G + 9.54}{47.3\,Cf}$$

Assuming the measured gain of 40 db we can compute the theoretical time delay

$$t_d = \frac{40 + 9.54}{(47.3)(.0286)10}10 = 3.65 \text{ m}\mu\text{sec}$$

The actual time delay as determined from helix dimensions and wave velocity is

$$t_d = \frac{\text{helix-length}}{\text{wave velocity}} = \frac{216 \times 10^{-3}}{205 \times 10^5} = 10.5 \text{ m}\mu\text{sec}$$

A certain amount of developmental work is therefore indicated before theoretical values of high gain coupled with short time delays are attained in traveling-wave tube amplifiers.

APPENDIX III

ARITHMETIC UNIT DESIGN

The devices and circuits described in Chapter IV are herewith combined into an arithmetic unit capable of algebraic addition, subtraction, and multiplication. Before embarking upon such a design numerous preliminary assumptions will be made in order to simplify the design procedure.

(1) Consideration will not be given at this point to the problems of selecting specific memory cells and transferring words from these cells to the arithmetic unit.

It will be assumed that words are initially available in a buffer memory or common memory bus ready for immediate transfer into the arithmetic unit registers.

(2) A word will be assumed to consist of 20 binary digits with the lowest order digit representing the sign, "zero" for plus and "one" for minus.

(3) Negative numbers will be stored in 1's complement form. Subtraction will be performed by addition of complements.

(4) Storage registers will be regenerative memory units with loop delay dependent upon operation being performed.

(5) Single bit registers with total loop delay of 2 μsec. to be used for sign bit storage.

(6) For illustrative purposes assume 1 μsec., 10 Kmc. pulses at a 500 mc. rate. Hence, bit time is 2 μsec. and word time is 40 μsec.

(7) Serial operation with adder accumulator.

REGISTER ASSIGNMENT:

B-Register - - - - For storage of Augend, Addend, Minuend, Subtrahend, and Multiplicand. Total loop delay of 1 word time (40 μsec.) for addition and subtraction, and 2 word times plus 1 bit time (82 μsec.) for multiplication.

C-Register - - - - For storage of Multiplier. Total loop delay of 2 word times minus 1 bit time (78 μsec.).

E-Register - - - - Sign bit storage for word in B-Register. Loop delay equals 1 bit time (2 μsec.).

F-Register - - - - Sign bit storage for word in C-Register. Loop delay equals 1 bit time (2 μsec.).

G-Register - - - - Multiplier bit storage. Loop delay equals 1 bit time (2 μsec.).

Accumulator Loop - For storage of Sum, Difference, and Product. For addition and subtraction, total loop delay of 1 word time (40 μsec.), for multiplication 2 word times (80 μsec.).

Outputs from the B and C Registers will be assumed to be available at times of (10 + ND) μsec. following the time of input, where N is any integer and D is the total loop delay. This expression is obtained from the assumption that the initial output can be obtained after the delay required for amplification, but before the signal traverses the entire loop delay. Successive outputs will then be obtained at intervals equal to the total loop delay. A delay of 10 μsec. is assumed for traveling-wave tube amplification.

In the case of the single bit registers, the travelling-wave tube will be assumed to have a delay of 1 μsec. and hence the output will be available at times of 1 + ND μsec. following the time of input.

PROCESS SCHEDULE

The process schedule which follows consists of a chronological listing of the actions required by the computer in accomplishing the various arithmetic operations. The symbol in the first column indicates the word time in which the indicated action will occur, while the small case letters appearing in the column to the right designate the control word which will be used to produce the designated action. Logical equations for the various control words will be developed from the requirements indicated by the process schedule. It should be kept in mind that although many separate actions may be listed during any particular word time these actions do not necessarily occur simultaneously. Because of the finite propagation time through the various logical elements, in particular the travelling-wave tube amplifiers, the commencing of any given word time at some specific point in the arithmetic unit will be delayed from the commencement of that word time at the arithmetic unit input by a time equal to the propagation time between the two points.

I. Transfer from memory bus to B-register. Instruction symbol - T

Time	Action	Control Word
To	During word time T_0 transfer contents of memory bus, R, into B-register. Complement if E = 1.	u
	Read out sign bit from word on memory bus and put it in E-register	v

II. Add (or Subtract) word on memory bus to word in accumulator. Assume that the accumulator has been previously filled with the

Augend (Minuend) as the result of a previous operation.

Instruction symbols: A - Add, S - Subtract.

Time	Action	Control Word
T_0	During word time T_0 send contents of memory bus to B-register and into adder accumulator through adder. Complement if instruction is subtract.	x
	Read out sign bit from word on memory bus and put it in E-register.	v
T_1	Recirculate accumulator contents to permit addition of end around carry.	
	Read out contents of accumulator and send to memory return bus.	x_d

III. Multiply word on memory bus times the contents of the B-register. Assume that the B-register is already filled and that the sign bit of the word in the B-register is in the E-register.

Time	Action	Control Word
	Set up B-register and accumulator for proper loop delay during Multiply time.	M
T_0	During word time T_0 send contents of memory bus to the C-register. Complement if F = 1.	y
	Read out sign bit and place in F-register.	w
T_0, T_2, T_4, ...	Read out lowest order operand digit from the C-register and put it in the G-register	z
.	During Multiply time, continuously add contents of B-register to accumulator if G = 1. If G = 0 add zero to contents of accumulator.	M

Time	Action	Control Word
T_{36}	During double word time $T_{36}T_{37}$ read out accumulator contents to memory return bus, complement if E or F (but not both) is 1.	t

CONTROL WORD EQUATIONS:

Equations for the required control words can now be written by reference to the process schedule and are given below. The symbols T, A, S, and M refer to the instructions Transfer, Add, Subtract and Multiply, respectively, while the timing is indicated by a symbol such as T_0 or $T_{0.0}$ where the subscript indicates the word time or word and digit times when the control word will equal 1. Hence, $T_{2.0}$ represents a signal that is equal to 1 during digit time zero of word time 2 and is zero at all other times.

(1) $t = M \cdot T_{36} \cdot T_{37}$

(2) $u = T \cdot T_0$

(3) $v = (T + A + S)\, T_{0.0}$

(4) $w = M \cdot T_{0.0}$

(5) $x = (A + S)\, T_0$

(6) $g = M \cdot T_0$

(7) $z = M \cdot T_{0.0} \cdot T_{2.0} \cdot T_{4.0} \cdot \ldots \cdot T_{36.0}$

A diagrammatic representation of the various timing signals required is shown in Figure 39. Zero time reference for these timing signals is the instant at which the first bit of a word on the memory bus is available for being read into the arithmetic unit. In the case where the designated control signals are used at other points within the arithmetic unit, appropriate delays must be included in the control signal path to insure proper arrival of the control signal relative to the operand bits.

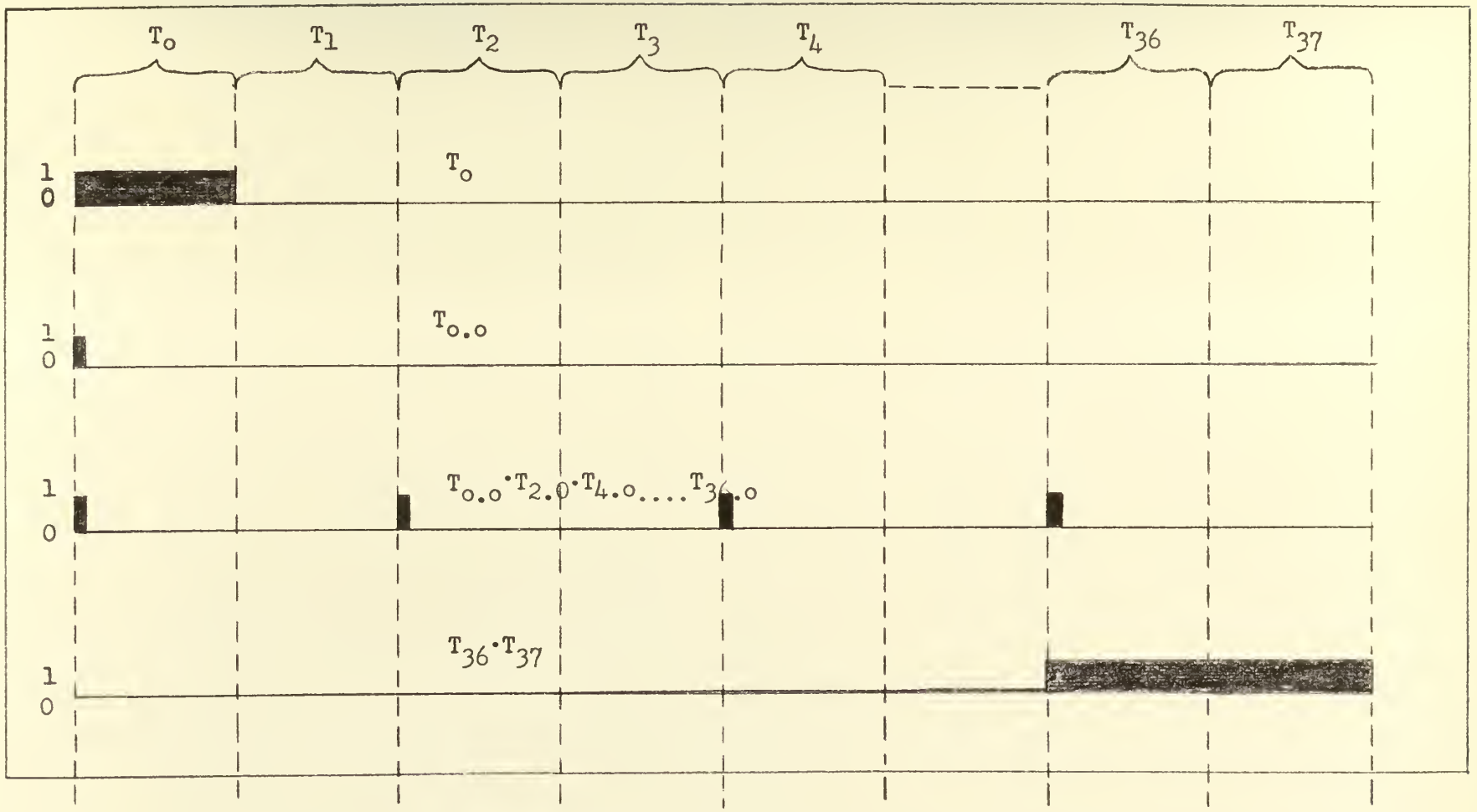


Fig. 39 - Diagrammatic representation of timing signals required in formation of control words

REGISTER INPUT EQUATIONS

The complete logic for the arithmetic unit can now be described in terms of the input equations for the five registers, the accumulator, and the memory bus as functions of register outputs and control words. In these equations capital letters refer to the output of the designated registers, primed capital letters refer to register inputs, Z refers to the accumulator and R to the memory bus.

The inputs to the E and F registers are given by equations (8) and (9) which provide for reading out the first (sign) bit of the word on the memory bus and setting up the E or F register accordingly.

(8) $E' = Rv$

(9) $F' = Rw$

Equation (10) provides for setting the G-register for a period of one word time according to the value of succeeding bits in the C-register.

(10) $G' = Cz$

The inputs to the B-register are described by equation (11).

(11) $B' = u(\overline{R}E + \overline{R}\overline{E}) + x(RA + \overline{R}S) + MB_{D_2} + \overline{M}\,B_{D_1} + B_D(\overline{u + x})$

Here, the first term provides for complementing the input during the transfer operation if the sign is negative. The second term provides for complementing if the instruction is Subtract. The third and fourth terms set up the proper total loop delay during Multiply time, and in these terms the symbols B_{D_1} and B_{D_2} represent the B-register output delayed by different amounts depending upon the operation being performed. The final term provides for erasing the circulating contents of the register during the time of any inputs and for recirculation when no inputs are present.

Equation (12) describes the input to the C-register.

(12) $C' = y(R\overline{F} + \overline{R}F) + \overline{y}\,C_D$

Here, again, the first term provides for complementing the input if its sign is negative and the last term provides for erasing and recirculation.

Equations (13) and (14) represent the two inputs to the accumulator.

$$(13)\quad Z_1' = MZ_{D_4} + \bar{M}Z_{D_3}$$

$$(14)\quad Z_2' = B(MG + x)$$

The first input equation provides for recirculation of the contents and proper choice of the total loop delay while the second input equation provides for input from the B-register during Add (Subtract) and Multiply instructions.

Equation (15) represents the input to the memory return bus. The

$$(15)\quad R' = Zx_d + \left[\bar{Z}(E\bar{F} + \bar{E}F) + Z\overline{(E\bar{F} + \bar{E}F)}\right] t$$

first term provides for input from the accumulator during Add or Subtract instructions. Here, the symbol $\mathbf{x}_d$ represents the control word "x" delayed by one word time which permits addition of the "end around carry." The second term provides for complementing the accumulator output before return to the memory during a Multiply instruction if E or F (but not both) is negative.

A schematic diagram of the logic for the complete arithmetic unit is shown in Figure 40. The relative timing of a signal at any point is indicated by its horizontal position relative to the input at the left side of the diagram to which point time is referenced. Thus, the control signals shown as being inserted at various points in the arithmetic unit must be delayed by the indicated amount in order to provide proper coincidence with the operand digits at each gate input. Clock signals are not shown in this diagram, but they, of course, must be present at all gates throughout the arithmetic unit with proper timing.

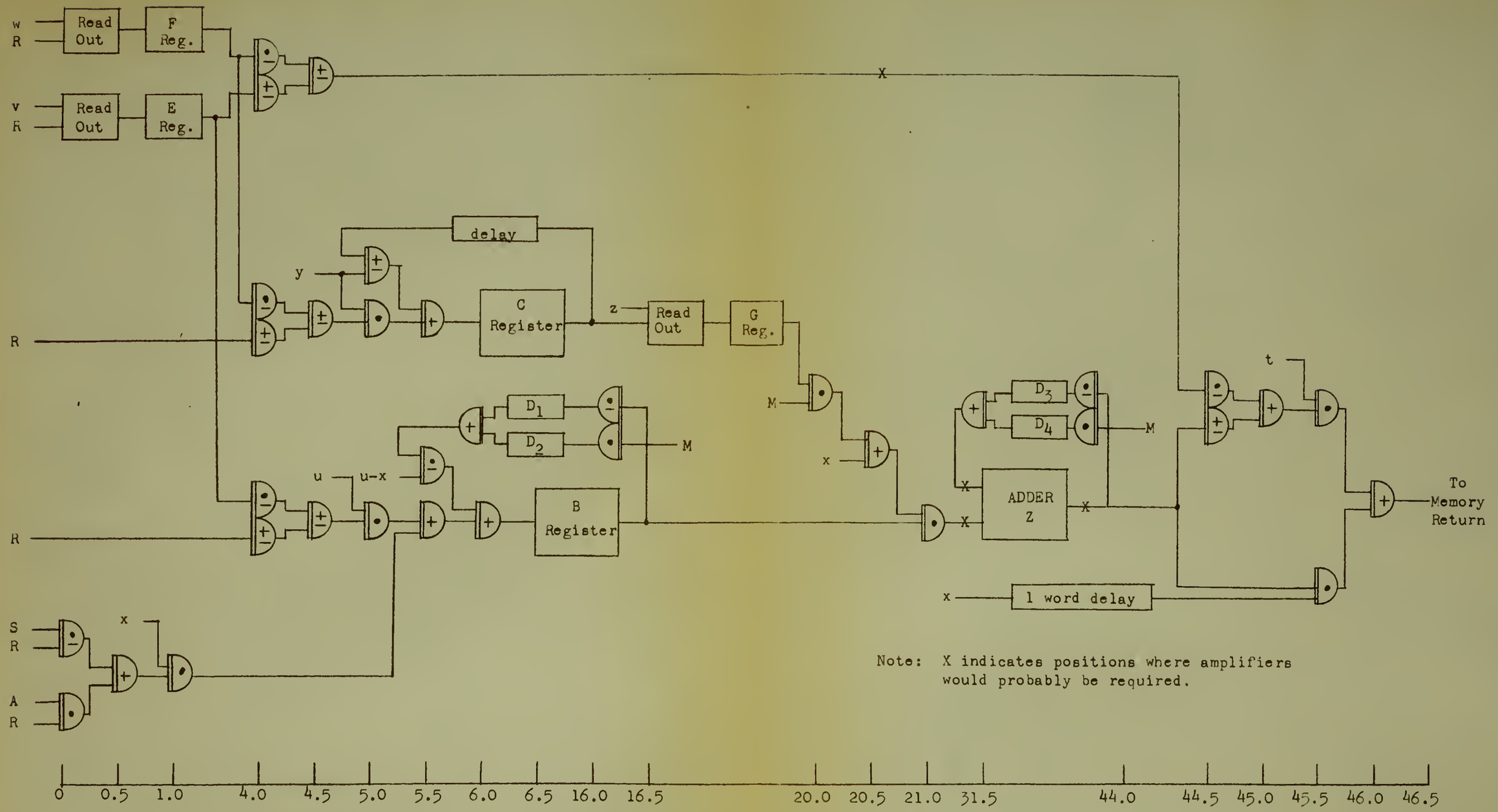


Figure 40 - Schematic Diagram of Phase Script Arithmetic Unit Logic